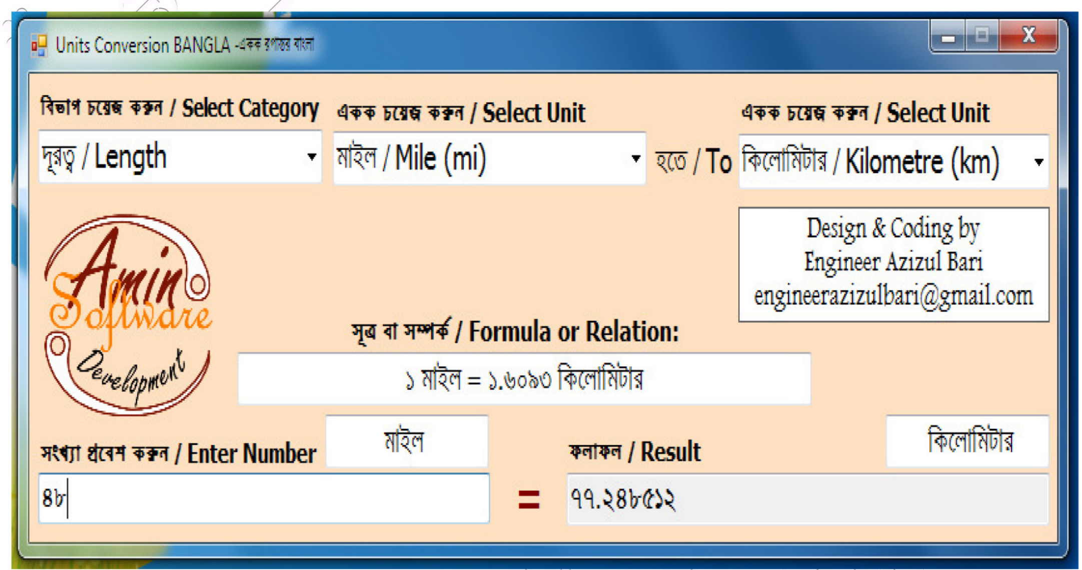
ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং

নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স

প্রথম খণ্ড

Part ONE



বিশেষ ইন্টারনেট সংস্করণ

Special Internet Edition

ইঞ্জিনিয়ার আজিজুল বারী
engineerazizulbari@gmail.com

খানকায়ে আমীনিয়া-আসগরিয়া গবেষণা বিভাগ
Ali Centre, Subidbazar, Sylhet, BD

বিষয়সূচি

বিষয়সূচি

১ম পরিচ্ছেদ: *ভূমিকা*

মাইক্রসফ্‌ট ভিজুয়্যাল বেসিক.নেট - কী এটা?

এটা একটি সহজ কম্পিউটার প্রোগ্রামিং ভাষা। যে কেউ ইচ্ছে, এ ভাষা শিখে বিস্ময়কর সফ্‌টওয়্যার তৈরি করতে পারে পি.সি কম্পিউটারের জন্য। এ ভাষার মৌলিক ক'টি বৈশিষ্ট্য:

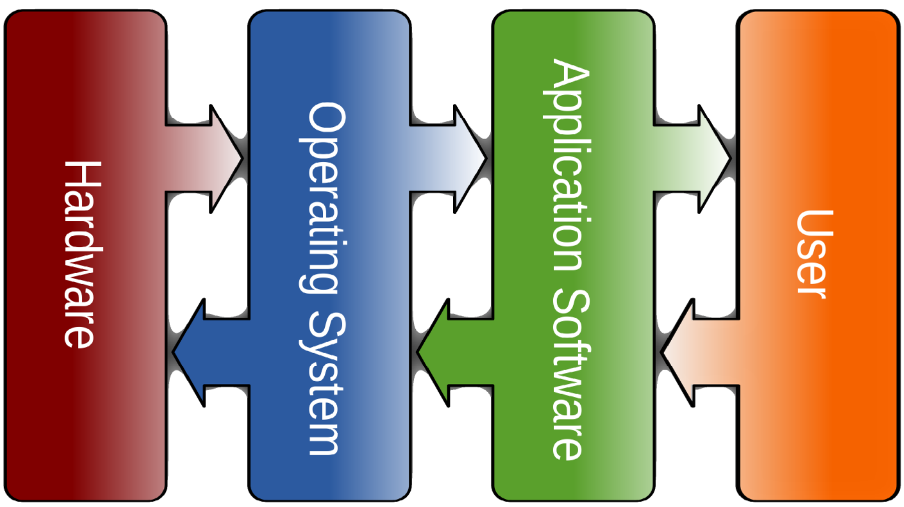
১. এটা বস্তুভিত্তিক প্রোগ্রামিং ভাষা (Object Oriented Programming Language)।
২. এটা আধুনিক, সহজ, জনপ্রিয়, ইংরেজির মতো ভাষা।
৩. এটা কাঠামোসর্বস্ব ভাষা (Structured Language)।
৪. এ ভাষায় লেখা প্রোগ্রামগুলো খুব নির্ভরশীল।
৫. ভুল-ত্রুটি সঠিককরণ (Debugging) অনেকটা সহজ।
৬. এটা মাইক্রসফটের ডট.নেট ফ্রেমওয়ার্ক (.Net Framework) এর অংশ।
৭. এ ভাষায় লিখিত সফ্‌টওয়্যার মাইক্রসফ্‌ট উইন্ডোজ এর জন্য খুব মানানসই।

জানেন কি?: পৃথিবীবিখ্যাত মাইক্রসফ্‌ট অফিস এর প্রতিটি এ্যাপলিকেশন ভিজুয়্যাল বেসিক (এবং ভিজুয়্যাল বেসিক.নেট) দ্বারা নিয়ন্ত্রণ ও রদবদল (customize) করা যায়।

এ পর্যন্ত যা বলা হলো তার এক বা একাধিক বিষয় আপনার বুঝে না আসলে ভাববেন না। এই গ্রন্থ ভালো করে অধ্যয়ন ও এতে সম্পৃক্ত বিভিন্ন পদে পদে বর্ণিত মডেল প্রোগ্রামগুলোর ওপর প্রাকটিস করে নিলে আপনিও হয়ে যাবেন ভিজুয়্যাল বেসিক.নেট প্রোগ্রামার। আপনাকে প্রোগ্রামার বানানোই আমাদের ইচ্ছে। পাঠ করতে থাকুন। দেখুন, কম্পিউটার প্রোগ্রামিং এর বিস্ময়কর জগৎ! এ বিস্ময়ের জগতে পাড়ি দিতে আপনার হাতিয়ার শুধুমাত্র একটি ল্যাপটপ বা ডেস্কটপ কম্পিউটার ও সেসাথে শেখার স্পৃহা- এই যা!

এই গ্রন্থে আপনি যা শিখবেন:

১. ভিজুয়্যাল বেসিক.নেট ভাষার মাধ্যমে সফ্‌টওয়্যার তৈরির পদ্ধতি।
২. উইন্ডোজের .exe এ্যাপলিকেশন তৈরি।
৩. উইন্ডোজের বেসিক ডাটাবেইজ প্রোগ্রামিং।



২য় পরিচ্ছেদ: শুরু করুন প্রাকটিস

কী লাগবে প্রাকটিস করতে?

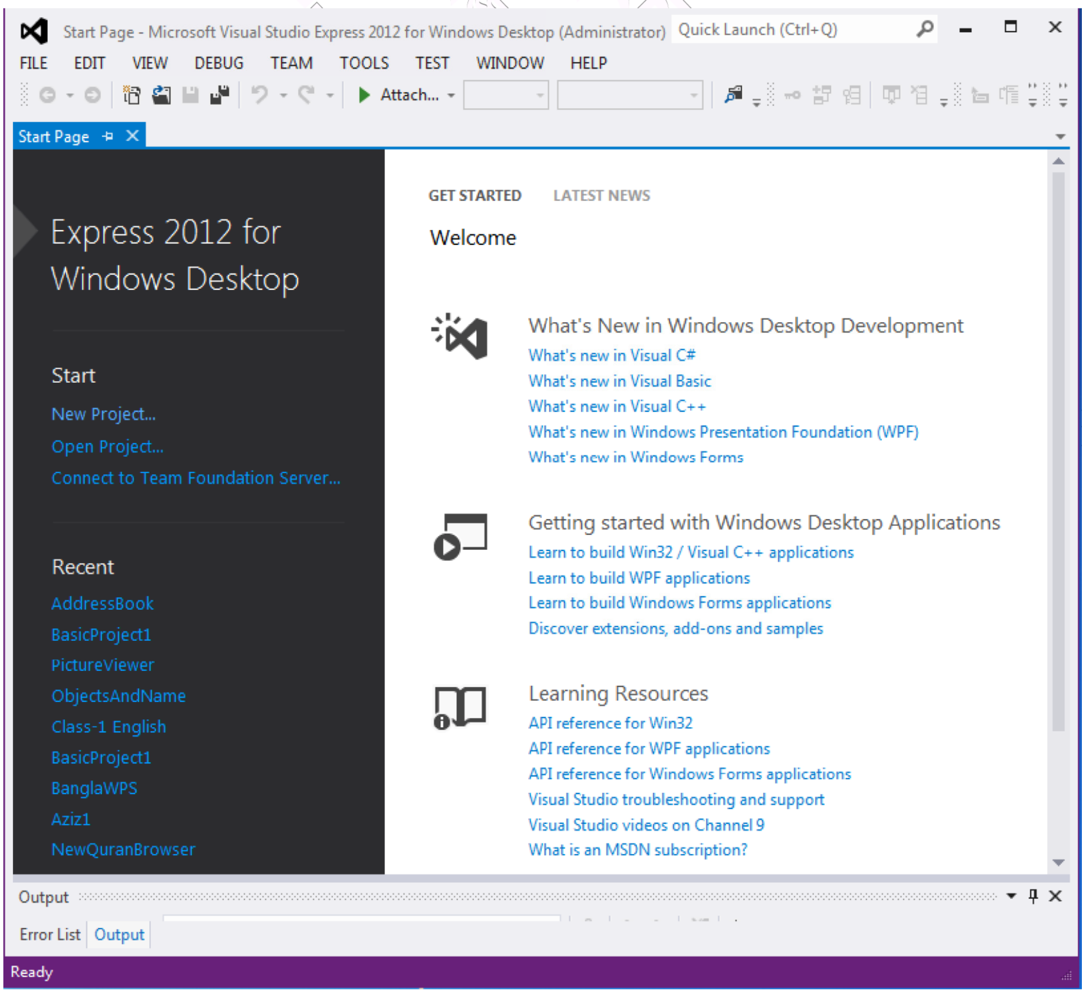
আই.ডি.ই (Integrated Development Environment)

বাংলায় বলা যায়, ‘অঙ্গীভূত উন্নয়ন পরিবেশ’। মূলত আই.ডি.ই হলো সফ্টওয়্যার তৈরির একটি সফ্টওয়্যার। এতে আছে বিভিন্ন ধরনের হাতিয়ার (Tools) যা ব্যবহারের মাধ্যমে আপনি সফ্টওয়্যার সৃষ্টি করতে সক্ষম হবেন। আপনার সুবিধার্থে মাইক্রসফ্ট একাধিক উন্নতমানের আই.ডি.ই তৈরি করেছে। আপনার পছন্দসই আই.ডি.ই ইন্টারনেট থেকে (অথবা কম্পিউটারের দোকান থেকে) কম্পিউটারে ইনস্টল করতে পারেন। এজন্য আপনাকে তেমন বেশি টাকা দিতে হবে না। ইন্টারনেট থেকে ফ্রি ডাউনলোড করতে পারেন।

এই গ্রন্থে আমি মা্ইক্রসফ্টের **‘ভিজুয়াল স্টুডিও এক্সপ্রেস ২০১২ ফর ডেস্কটপ’** থেকে উদাহরণ দিয়েছি। আপনি এটা ডাউনলোড করতে পারেন এই ওয়েভ ঠিকানা থেকে:

https://www.microsoft.com/en-us/download/details.aspx?id=34673

ইনস্টল করার পর মাইক্রসফ্টের ওয়েভসাইট থেকে বিনামূল্যে রেজিস্ট্রেশন করুন। এরপর ভিজুয়াল বেসিক এক্সপ্রেস ফর ডেস্কটপ যখনই রান করবেন তখন প্রথমে এই উইনডোটি আসবে:



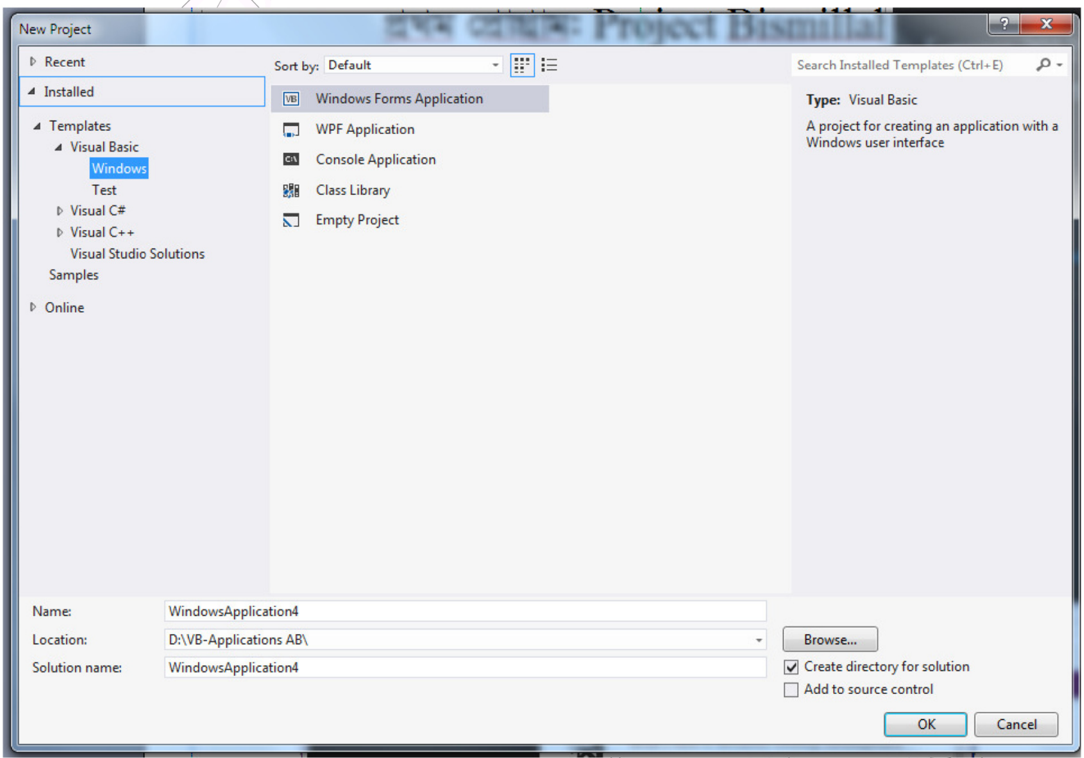
চিত্র-১: এক্সপ্রেস ২০১২ ফর উইন্ডোজ ডেস্কটপ রান করার পর প্রথমে প্রদর্শিত উইন্ডো।

১ম প্রোগ্রাম: ProjectBismillah

নিচে ধারাবাহিকভাবে বর্ণিত নির্দেশনাবলী অনুসরণ করে প্রজেক্ট তৈরি করুন:

১. Start এর নিচে লিখিত New Project... -এ ক্লিক করুন।

নতুন একটি প্রজেক্ট তৈরির জানালা আসবে। এটি নিম্নে দেখানো হলো।



চিত্র-২: New Project... -এ ক্লিক করার পর প্রদর্শিত জানালা।

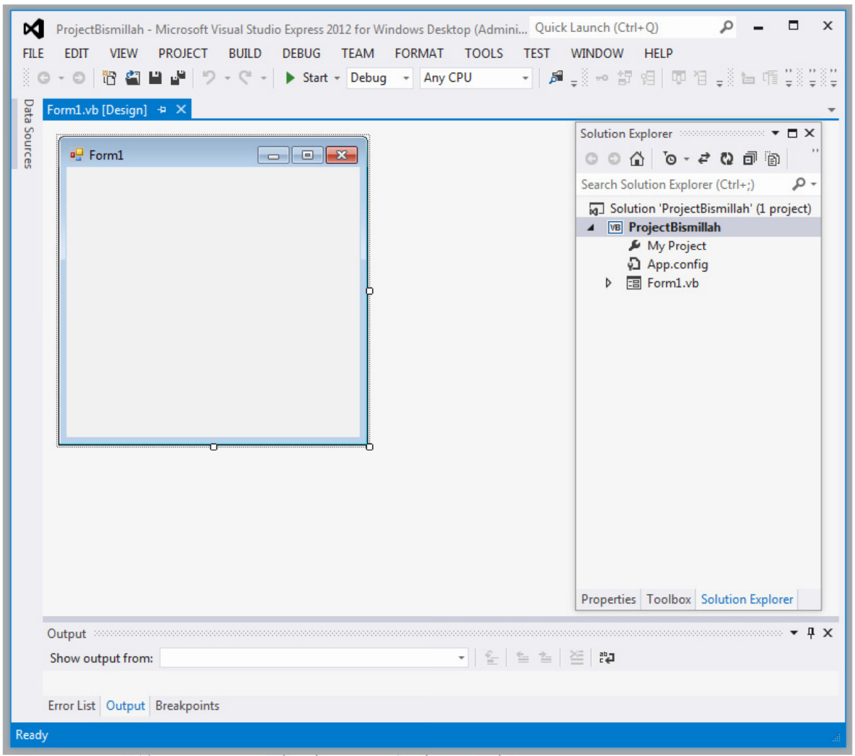
আপনি ডেস্কটপে LearnigDotNet নামকরণে একটি ফন্ডার বানিয়ে এর ভেতর সকল Project সংরক্ষণ করতে পারেন।

২. নতুন প্রজেক্ট ProjectBismillah -সংরক্ষণ করুন।

জানালার Name ফিল্ডে লিখুন ProjectBismillah। Windows Forms Application যেনো হাইলাইট (ক্লিক করা) থাকে। আমরা এই গ্রন্থে এরূপ প্রোগ্রামই করবো। Browse...-এ ক্লিক করে আপনার পছন্দসই ফন্ডারে সংরক্ষণের জন্য চয়েজ করুন। এরাব ক্লিক করুন 'Ok' বোতামে। আপনার নতুন প্রজেক্ট তৈরি হবে। কিছুক্ষণ পর প্রজেক্ট খুলবে এবং পরের পৃষ্ঠায় চিত্রিত জানালাটি প্রদর্শিত হবে।

ব্যস! আপনি প্রথম প্রজেক্ট তৈরি করে নিলেন। এবার আসুন, এই প্রজেক্টের উদ্দেশ্য হাসিলে কাজে লেগে যাই- কেমন?

১ম প্রোগ্রাম: ProjectBismillah



চিত্র-৩: ProjectBismillah তৈরির পর প্রদর্শিত আই.ডি.ই জানালা।

উপরের চিত্রের মতো আপনার কম্পিউটারে প্রদর্শিত আই.ডি.ই ঠিক হুবহু একই না-ও হতে পারে। কোনো ভাবনা নেই। যেভাবেই আসুক না কেনো, এতে সর্বদাই Form1 নামকরণে একটি ফরম থাকবে। এই ফরম বা জানালাটির ভেতরই আমাদের প্রোগ্রামের যাবতীয় বস্তু তৈরি হবে। একটু পরই আপনি বিষয়টি বুঝতে সক্ষম হবেন।

৩. সলিউশন এক্সপ্লোরার প্রদর্শন করুন।

যদি Solution Explorer জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে চতুর্থ আইটেম Solution Explorer চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন।

৪. টুলবক্স প্রদর্শন করুন।

যদি Toolbox জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে ত্রয়োদশ আইটেম Toolbox চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন। চাইলে Solution Explorer জানালার ভেতর ছেড়ে দিতে পারেন।

৫. প্রোপার্টিজ প্রদর্শন করুন।

যদি Properties জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে শেষেরটির আগের আইটেম Properties Window চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন। চাইলে Solution Explorer জানালার ভেতর এটাও ছেড়ে দিতে পারেন।

উপরে বর্ণিত উপায়ে কাজ করলে যে জানালাটি দেখা যাবে তা হবে এ পৃষ্ঠার চিত্রের মতো (চিত্র-৩)। লক্ষ করুন Solution Explorer জানালার নিচে তিনটি বোতাম আছে: Properties, Toolbox এবং Solution Explorer। যে সময় যেটির প্রয়োজন তাতে ক্লিক করলেই সে জানালা প্রদর্শিত হবে। আপাতত আমাদের দরকার এই তিনটি ও Form1 নামক জানালাটি।

ব্যস! ProjectBismillah -এর জন্য আপনি আই.ডি.ই প্রস্তুত করে নিয়েছেন। এবার কিছু প্রোগ্রামিং হবে।

১ম প্রোগ্রাম: ProjectBismillah

User Interface ডিজাইন

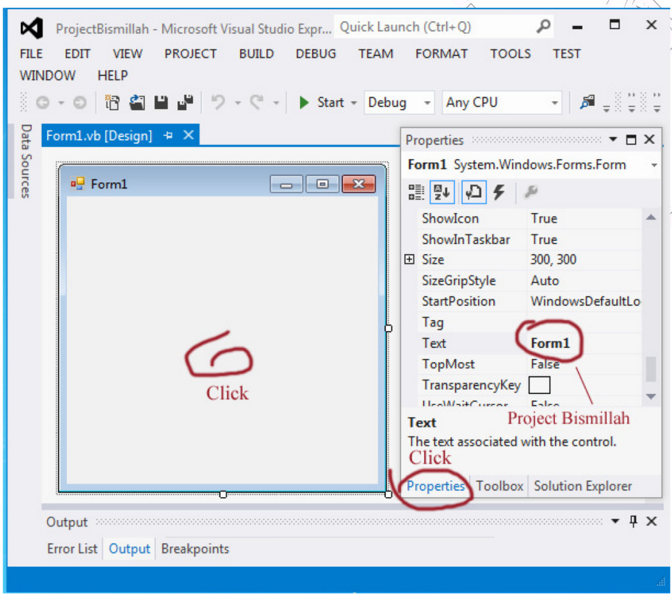
বাংলায় ব্যবহারকারীর সাথে প্রোগ্রামের মিথস্ক্রিয়াকে ইউজার ইন্টারফেইস বলা যায়। কিভাবে একজন ব্যবহারকারী আপনার প্রোগ্রাম থেকে কাঙ্ক্ষিত ফলাফল বা সুবিধা উপভোগ করবে তা সুন্দরভাবে ডিজাইন করার নামই হলো ইউজার ইন্টারফেইস। বিসমিল্লাহ প্রোগ্রামে আমরা যা চাচ্ছি তাহলো, একটি বোতামে ক্লিক করলে ইংরেজিতে 'Bismillahir Rahmanir Raheem' বাক্যটি একটা ফিল্ডে লেখা হবে। একই সময় একটি ডায়লোগ বা মেসেজ বাক্স আসবে ও "Beep" শব্দ হবে। বাক্সে ইংরেজিতে লেখা থাকবে "In the name of Allah the Merciful, the Beneficent." - এ কথাটি।

এ ইন্টারফেইস তৈরিতে আমাদের যা দরকার:

১. একটি প্রোগ্রাম ফরম বা উইন্ডো।
২. একটি লেখা প্রদর্শনের ফিল্ড বা টেক্সট্ বাক্স।
৩. একটি ক্লিক করার বোতাম।

উপরে উল্লেখিত তিনটি বস্তুর মধ্যে প্রথমটি তথা উইন্ডো ফরম ইতোমধ্যে আমরা তৈরি করে নিয়েছি। চাইলে একে সামান্য পরিবর্তন করতে পারি। যেমন, এর উপরের শিরোনাম '**Form1**' পরিবর্তন করে আমরা লিখতে পারি '**Project Bismillah**'। এটা করতে নিচের ধারাবাহিক পদক্ষেপগুলো গ্রহণ করুন।

১. ফর্মের ভেতর ক্লিক করে একে সিলেক্ট করুন। ২. সলিউশন এক্সপ্লোরারের নিচে 'Properties' শব্দের উপর ক্লিক করুন। ৩. স্ক্রল করে 'Text' আইটেমে যেয়ে 'Form1' শব্দ মুছে সেখানে লিখুন 'Project Bismillah'। এবার সেইভ করুন (Ctrl+Shift-S কিংবা File - Save All)। নিচের চিত্রটি (চিত্র-৪) দেখুন।



চিত্র-৪: Form1 -এর শিরোনাম পরিবর্তন।

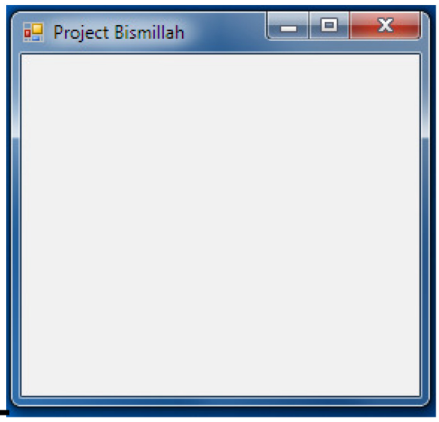
প্রোগ্রাম "Start" করা

যে কোনো প্রোগ্রাম তৈরির সময় বার বার এটি শুরু করে কাজ করছে কি না দেখতে হয়। এ সুবিধার জন্য "Start" নামকরণে একটি মেনু আইটেম দেওয়া আছে। এতে ক্লিক করলে প্রোগ্রাম বিল্ড করবে। প্রোগ্রামের কোডিংয়ে ভুল থাকলে সতর্ক মেসেজ প্রদর্শিত হবে। অন্যথায় সফলভাবে প্রোগ্রাম শুরু হবে। আপনি কাঙ্ক্ষিত ফলাফল পাচ্ছেন কি না পরীক্ষা করে নিতে পারেন। এরপর এটা বন্ধ করলেই আই.ডি.ই -তে ফিরে আসবেন।

এ পর্যন্ত আমরা যা করেছি তা দেখার জন্য একবার "Start" বোতামে ক্লিক করুন। নিচের চিত্র-৫ এর মতো একটি উইন্ডো দেখা যাবে।

মনে রাখবেন:

কিছুক্ষণ কাজ করেই **Ctrl+Shift-S** কিংবা **File - Save All** দ্বারা প্রজেক্ট সেইভ বা সংরক্ষণ করবেন। এভাবে সংরক্ষণের অভ্যেস করে নেবেন। আপনার কাজ অনাকাঙ্ক্ষিত কোনো কারণে নষ্ট হবে, তা-তো মানা যায় না!



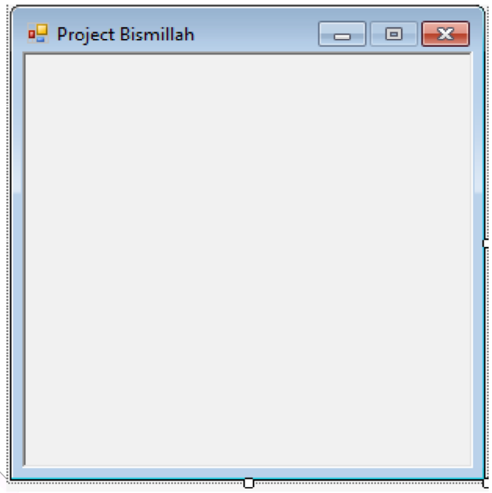
চিত্র-৫: Start -করার পর প্রদর্শিত জানালা

১ম প্রোগ্রাম: ProjectBismillah

FormBorderStyle ও MaximumBox ডিজাইনে পরিবর্তন

প্রোগ্রাম চলার সময় যাতে ফরম জানালার আয়তন পরিবর্তন না হয় সে জন্য দু'টি প্রোপার্টিতে পরিবর্তন করলে ভালো হবে। এই দু'টি পরিবর্তন করতে নিম্নের ধারাবাহিক পদক্ষেপগুলো অনুসরণ করুন:
ফর্মে ক্লিক করে এটাকে সিলেক্ট করুন। ক্লিক করুন সলিউশন এক্সপ্লোরার জানালায়। এবার স্ক্রল করে FormBorderStyle -এ যান। ডানের ড্রপডা্উন মেনু থেকে চয়েজ করুন Fixed3D আইটেম। এরপর স্ক্রল ক ে র চলে যান MaximumBox -এ। এর ডানের বাক্সে True লেখা থকালে ড্রপডাউন বাক্স থেকে False শব্দটি চয়েজ করুন।

এবার আপনার ফর্মের চেহারা ডানের চিত্রের মতো হবে। প্রোগ্রাম Start করে দেখুন- ফর্মের সাইজ পরিবর্তন করা আর সম্ভব হবে না।



চিত্র-৬: Form1 -এর চিত্র।

একটি Button ও একটি Text box ফর্মে যুক্তকরণ

আমরা ফর্মের ডিজাইন শেষ করেছি। এবার এর ভেতর দু'টি বস্তু যুক্ত করবো। প্রথমে একটি Text box ফর্মের উপরে এনে বসাবো। নিচে বর্ণিত পদক্ষেপগুলো ধারাবাহিকভাবে অনুসরণ করুন।
ক্লিক করুন সলিউশন এক্সপ্লোরার জানালায়। ক্লিক করুন নিচের Toolbox বাক্সে। Common Controls -এর নিচে দেখতে পাবেন অনেকগুলো আইটেম। ডবল ক্লিক করুন ২নং আইটেম Button-এ। দেখবেন, Button1 -নামক একটি বোতাম ফর্মের উপরের বায়ের কোণে চলে গেছে। আপনি বোতাম যোগ করে নিয়েছেন।

এবার টুল বাক্সে যেয়ে স্ক্রল করে নিচে যান। আগের মতো ডবল ক্লিক করুন TextBox -নামক আইটেমে। লক্ষ করুন, একটি বাক্স চলে গেছে ফর্মের ভেতর। এটাকে মাউস দিয়ে টেনে মাঝখানে নিয়ে আসুন। বোতামটিও নিচে সেন্টারে নিয়ে আসুন।

টেক্সট্ বাক্স ও বোতামের প্রোপার্টি পরিবর্তন

আমরা চাই টেক্সট্ বাক্সের লেখাটুকু একটু বড় সাইজ হবে। এছাড়া বোতামে লেখা থাকবে Click Me শব্দগুলো। লেখার সাইজও কিছুটা বড় হবে। আমরা এ কাজগুলো সহজে করতে পারি প্রোপার্টিজ জানালা ব্যবহার করে। নিম্নে বর্ণিত পদক্ষেপগুলো অনুসরণ করুন।

Button - ক্লিক করে Button1 সিলেক্ট করুন। এক্সপ্লোরার জানালার নিচে Properties বাক্সে ক্লিক করুন। এবার প্রথম দিকের + Font -এর + চিহ্নে ক্লিক করুন। আরো একটি মেনু আসবে। এখানে Size কথাটির বাক্সে ৮ মুছে ১৪ লিখুন। প্রোপর্টিজ জানালার ডানের স্ক্রলে ধরে নিচের দিকে যেয়ে Text আইটেম বের করুন। এর ডানে Button1 শব্দটি মুছে Click Me শব্দদ্বয় লিখুন। ব্যস! বোমাতের পরিবর্তন সম্পূর্ণ হলো।

TextBox - ক্লিক করে টেক্সট্ বাক্স সিলেক্ট করুন। প্রোপার্টিজ জানালায় যেয়ে আগের মতো ফন্ট সাইজ বদলে ১৪ করুন। ব্যস! হয়ে গেল।

ফর্মের সাইজ পরিবর্তন, টেক্সট্ বাক্স ও বেতামের অবস্থান পরিবর্তন শেষে যে ফর্ম তৈরি হলো তা নিম্নে দেখানো হয়েছে।



চিত্র-৭: Form1 -এর শেষ চিত্র।

১ম প্রোগ্রাম: ProjectBismillah

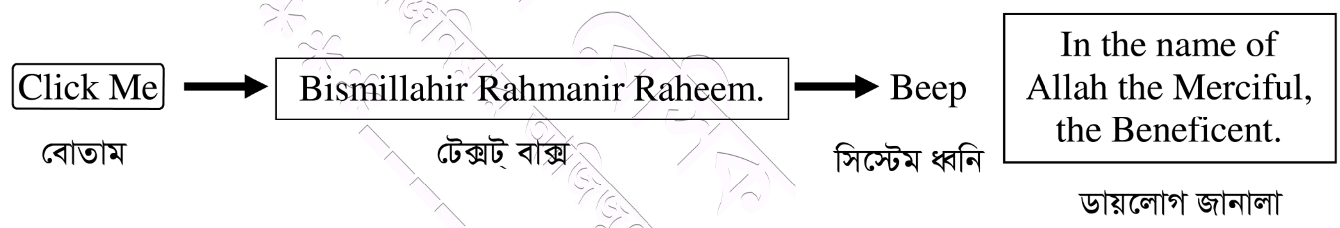
Designer থেকে Programmer

এ পর্যন্ত আপনি যা করলেন তাতে আপনাকে প্রোগ্রামার উপাধিতে ভূষিত করা যাবে না! হ্যাঁ, আপনি ডিজাইনার ছিলেন। কারণ আপনি একটি ইউজার ইন্টারফেইস ডিজাইন করেছেন সফলভাবে। তবে আমাদের ইচ্ছা আপনি শুধু ডিজাইনার নয় একজন দক্ষ প্রোগ্রামার হবেন। অর্থাৎ বুঝেসুঝে আপনি প্রোগ্রাম চলাকালের এ্যাকশন কোড লিখার যোগ্যতা লাভ করবেন। আর এটাই তো এ গ্রন্থ রচনার উদ্দেশ্য।

প্রোগ্রাম করা যেমন কঠিন মনে করা হয় বাস্তবে তা কিন্তু নয়। অবশ্য এজন্য মাইক্রসফটকে ধন্যবাদ দিতেই হবে। এক সময় এমন ছিলো, আপনি যে ইউজার ইন্টারফেইস কয়েক মিনিট বা ধরুণ ঘণ্টা খানেকের মধ্যেই তৈরি করে নিলেন তা করতে, লিখতে হতো শত শত লাইনসর্বস্ব কোড। সময় লাগতো কয়েক দিন পর্যন্ত। কিন্তু এ কাজগুলো আমরা রেডি-মেইড পেয়ে যাচ্ছি। যা তারা রেডি-মেইড করে দিতে পারেন নি তাহলো আমাদের চাহিদা। কারণ তারা তো জানেন না, প্রোগ্রাম দ্বারা আমরা কি করতে চাই? এ কারণেই বাকী অংশ নিজেরাই করতে হবে।

প্রোগ্রাম কোডিং

আমরা চাই, যে মুহূর্তে ব্যবহারকারী Click Me বোতামে ক্লিক করবেন, তখনই কিছু তথ্য প্রদর্শন করতে। কিসব তথ্য প্রদর্শিত হবে তা নিম্নে চিত্রিত হয়েছে।



উইন্ডোজ ফরম এপলিকেশনের কোডিং ফিল্ডে সর্বদাই নিম্নে চিত্রিত একটি কোড ব্লক থাকে। এই ব্লকের ভেতরেই যাবতীয় কোড লিখতে হবে।

```
Public Class Form1

   সব কোড এখানে

End Class
```

Click Me - বোতামের কোড

ডিজাইন উইন্ডোতে চিত্রিত বোতামে ডবল ক্লিক করলেই নিচের ফিল্ডটি আসবে। আমাদের লেখা লাইনটি আন্ডারলাইন করে দেখানো হয়েছে। মূল কোডে আন্ডারলাইন থাকবে না।

```
Public Class Form1

   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
         TextBox1.Text = "Bismillahir Rahmanir Raheem."
   End Sub

End Class
```

এই `TextBox1.Text = "Bismillahir Rahmanir Raheem."` লাইন কোড দ্বারা আমরা নির্দেশ দিচ্ছি টেক্সট বাক্সে যা লিখা হবে।

১ম প্রোগ্রাম: ProjectBismillah

আমরা এবার Click Me বেতামের Sub-এ আরো দু'টি লাইন যুক্ত করে নেবো। লাইন দুটো হলো:

```
Beep()
MsgBox("In the name of Allah the Merciful, the Beneficent.")
```

ব্যস! কোডিং শেষ হলো। Form1.vb* -নামক ফিল্ডে এবার যা লিখা থাকবে তাহলো:

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      TextBox1.Text = "Bismillahir Rahmanir Raheem." 'what we want to be written in Text Box.
      Beep() 'System sound.
      MsgBox("In the name of Allah the Merciful, the Beneficent.") 'The text for message box window.
   End Sub
End Class
```

ব্যাখ্যা: Click Me বোতামের Click নামক একটি Event আছে। আমরা এই ইভেন্টটি ধরে কম্পিউটারকে ৩টি কোড লাইনে নির্দেশ দিচ্ছি। যথা: লাইন-১ দ্বারা বলা হচ্ছে ফর্মের TextBox1-এ কী লেখা হবে। লাইন-২ দ্বারা নির্দেশ দিচ্ছি কম্পিউটারের Beep নামক ধ্বনিটি প্লে করার জন্য। এটি সিস্টেমে সংরক্ষিত আছে। লাইন-৩ দ্বারা বলা হচ্ছে একটি মেসেজ বাক্স খুলে তাতে কিছু লিখে প্রদর্শন করতে।

দ্র: লক্ষ করুন, কোডে সবুজ কিছু লেখা আছে। এগুলোর শুরুতে ' -এ সিঙ্গুল কোট মার্কও দেওয়া হয়েছে। এ সবুজ লাইনগুলো হলো 'কমেন্ট'। ভিজুয়্যাল বেসিক এগুলো সম্পূর্ণরূপে উপেক্ষা করে যাবে। প্রোগ্রামারের সুবিধার্তে এই কমেন্ট লাইনের ব্যবস্থা। মন্তব্য করলে কোডের অর্থ অনেকটা স্পষ্ট হয়ে ওঠে। আপনিও অবশ্যই মন্তব্য করবেন। এগুলো রেডি-হওয়া প্রোগ্রামের কোনো ক্ষতি করে না কিংবা এদের কারণে অতিরিক্ত মেমোরিও খরচ হয় না। তবে প্রয়োজনে প্রোগ্রাম সঠিককরণের [ডিবাগিং] সময় এসব মন্তব্য বিরাট উপকারী হবে। মনে রাখবেন, কোনো কোনো প্রোগ্রামের কোডের লাইন সংখ্যা শত শত থেকে হাজার হাজারে যেয়ে পৌঁছতে পারে!

১ম প্রোগ্রাম শেষ!

আপনি সফলভাবে প্রথম প্রোগ্রাম শেষ করেছেন। তবে আই.ডি.ই বন্ধ করার পূর্বে অবশ্যই একবার প্রোগ্রাম টেস্ট করে দেখতে হবে। নিচের চিত্র-৮-এ এই টেস্টের ফলাফল চিত্রিত হলো।



চিত্র-৮: ১ম প্রোগ্রাম ProjectBismillah পরিচালনা করে Click Me বোতামে ক্লিক করার ফলাফল।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

ভিজুয়্যাল বেসিক.নেট : Subs & Functions

ডট.নেট প্রোগ্রামের কোড মূলত Sub [Subroutine] ও Function নিয়ে গঠিত। ১ম প্রোগ্রামে আমরা Sub ব্যবহার করেছি। এ দু'টো হলো প্রোগ্রাম কোডের মধ্যে এককেকটি অংশ।

প্রত্যেক Sub শুরু হয় এভাবে [এর Syntax] :

```
[accessibility] Sub routine_name [paramaters]
    statements
End Sub
```

accessibility = Public, Protected, Friend, Protected Friend, or Private

Public -অর্থ এই সাবরুটিন সমগ্র প্রোগ্রামব্যাপী ব্যবহারযোগ্য।

Private -অর্থ যে মডিউলের ভেতর এটি আছে শুধু সেখানেই ব্যবহারযোগ্য।

আপাতত এ দু'টোই আমরা ব্যবহার করবো।

routine_name -যে কোনো নাম। নামটি ভিজুয়্যাল বেসিকের আইন মুতাবিক হতে হবে। যে কোনো অক্ষর বা আন্ডারস্কোর [_] দ্বারা নাম শুরু হবে। নামের মধ্যে কোনো স্পেইস [শূন্যস্থান ক্যারাক্টার] থাকবে না।

paramaters - ঠিক ভেরিয়েবল্ ঘোষণার মতো। একটু পর কিছু উদাহরণ দ্বারা সব পরিষ্কার হবে।

প্রত্যেক Function শুরু হয় এভাবে [এর Syntax]:

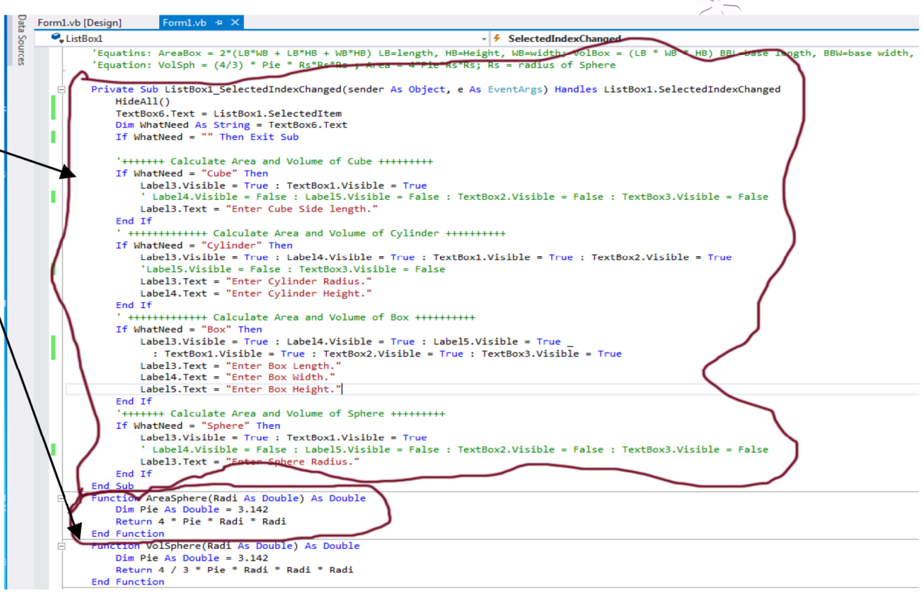
```
[accessibility] Function function_name [paramaters] [As return_type ]
    statements
End Function
```

প্রতিটি আইটেম Sub ঘোষণার অনুরূপ। আর As return_type -এর অর্থ হলো কোন্ ধরনের তথ্য ফাংশন ফিরে দেবে।

মনে রাখবেন: প্রত্যেক ফাংশনের কাজ হলো কোনো তথ্য [ভ্যালু] ফিরে দেওয়া।

বিষয়টি স্পষ্টভাবে বুঝার জন্য নিচের কোড অংশ দেখুন (চিত্র-৯)। এই কোডিংয়ের উদাহরণ পরে আসছে।

চিত্র-৯: Sub এবং Function লেখা হয়েছে ফর্মের ক্লাস ব্লকের ভেতর।



```
'Equatins: AreaBox = 2*(LB*WB + LB*HB + WB*HB) LB=length, HB=Height, WB=width; VolBox = (LB * WB * HB) BBL=base length, BBW=base width,
'Equation: VolSph = (4/3) * Pie * Rs*Rs*Rs ; Area = 4*Pie*Rs*Rs; Rs = radius of Sphere

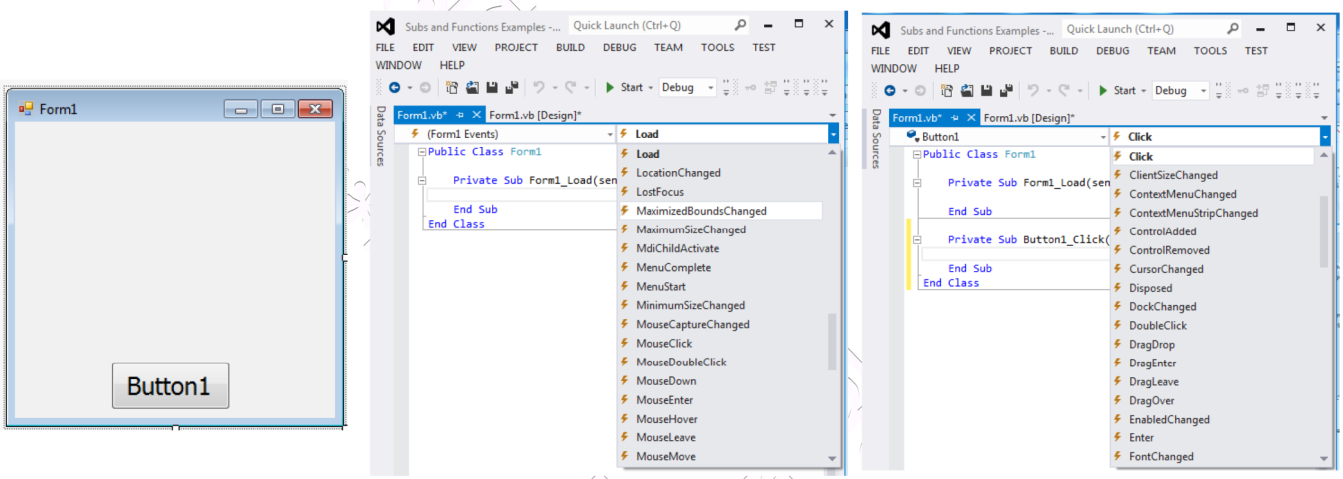
Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged
    HideAll()
    TextBox6.Text = ListBox1.SelectedItem
    Dim WhatNeed As String = TextBox6.Text
    If WhatNeed = "" Then Exit Sub

    '+++++++ Calculate Area and Volume of Cube +++++++++
    If WhatNeed = "Cube" Then
        Label3.Visible = True : TextBox1.Visible = True
        ' Label4.Visible = False : Label5.Visible = False : TextBox2.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Cube Side length."
    End If
    ' +++++++++++++ Calculate Area and Volume of Cylinder ++++++++++
    If WhatNeed = "Cylinder" Then
        Label3.Visible = True : Label4.Visible = True : TextBox1.Visible = True : TextBox2.Visible = True
        'Label5.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Cylinder Radius."
        Label4.Text = "Enter Cylinder Height."
    End If
    ' +++++++++++++ Calculate Area and Volume of Box ++++++++++
    If WhatNeed = "Box" Then
        Label3.Visible = True : Label4.Visible = True : Label5.Visible = True _
          : TextBox1.Visible = True : TextBox2.Visible = True : TextBox3.Visible = True
        Label3.Text = "Enter Box Length."
        Label4.Text = "Enter Box Width."
        Label5.Text = "Enter Box Height."
    End If
    '+++++++ Calculate Area and Volume of Sphere +++++++++
    If WhatNeed = "Sphere" Then
        Label3.Visible = True : TextBox1.Visible = True
        ' Label4.Visible = False : Label5.Visible = False : TextBox2.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Sphere Radius."
    End If
End Sub
Function AreaSphere(Radi As Double) As Double
    Dim Pie As Double = 3.142
    Return 4 * Pie * Radi * Radi
End Function
Function VolSphere(Radi As Double) As Double
    Dim Pie As Double = 3.142
    Return 4 / 3 * Pie * Radi * Radi * Radi
End Function
```

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

১ম উদাহরণ: Sub

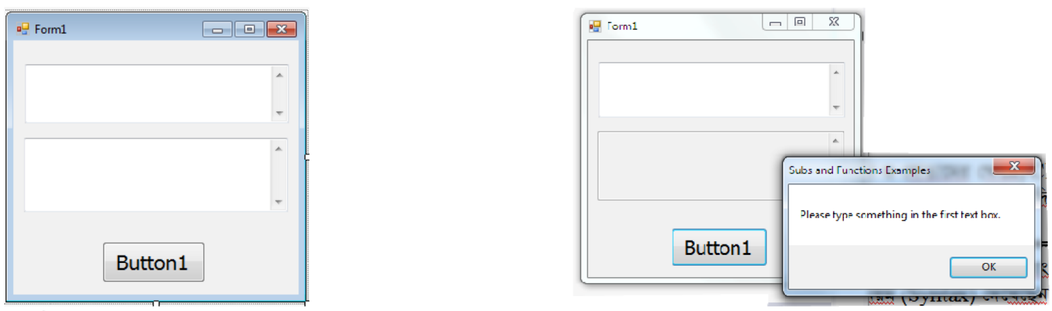
Sub এর উদাহরণ উপস্থাপনের পূর্বে আপনাকে একটি বিষয় খুব ভালো করে জানতে হবে। Form, Button, TextBox, Label, ListBox, ComboBox ইত্যাদি টুলবাক্সের যাবতীয় বস্তুর [Objects] একাধিক event [ঘটনা] আছে। প্রোগ্রামার হিসাবে আপনার কাজ হলো এসব ঘটনাকে Sub -এর মধ্যে ধরে নিজের ইচ্ছে মতো কাজ করানো। নিচের চিত্রে [চিত্র-১০] ফরম ও বোতামের কিছু ঘটনা [events] দেখানো হয়েছে। এসব ইভেন্ট কোড উইন্ডোতে যেয়ে সিলেক্ট করলেই একেকটি Sub আই.ডি.ই বানিয়ে দেবে। আপনি এর ভেতর কোড লিখে ইভেন্টকে নিয়ন্ত্রণ ও কাজে লাগাবেন। আর এভাবে ইভেন্ট কাজে লাগানোর নামই প্রোগ্রামিং। আপনি যদি সফলভাবে এগুলো কাজে লাগাতে সক্ষম হন তাহলে আপনি হবেন একজন দক্ষ প্রোগ্রামার।



চিত্র-১০: (বায়ে) একটি Project। (মাঝে) প্রজেক্টের ফর্মের ইভেন্ট লিস্ট। (ডানে) প্রজেক্টের বাটন১ এর ইভেন্ট লিস্ট।

অধিকাংশ ক্ষেত্রে আপনি যেসব ইভেন্ট কাজে লাগাবেন তাহলো: (ফরম ও বোতামের ক্ষেত্রে) Click, MouseUp, MouseDown, MouseEnter, MouseLeave; (লিস্টবাক্স, কম্বোবাক্স, টেক্সটবাক্স ইত্যাদির ক্ষেত্রে) Click, MouseEnter, TextChanged, MouseLeave ইত্যাদি।

Sub শব্দের অর্থ Subroutine। ভিজুয়্যাল বেসিক.নেট -এ সাব -এর গুরুত্ব অপরিসীম। সুতরাং সাব কিভাবে কাজ করে তা আপনাকে জানতেই হবে। আপনি আগের পৃষ্ঠায় সাব লেখার নিয়ম (Syntax) দেখেছেন। পরের পৃষ্ঠার সাব দু'টি উপরের চিত্র-১০ -এ চিত্রিত বোতামের Click event ও টেক্সট বাক্সের MouseLeave event সাব, যা আমরা লিখেছি দু'টি কাজ আঞ্জাম দিতে। উক্ত ফর্মে এবার দু'টি TextBox কন্টোলও যোগ করেছি। নিচের বায়ের চিত্রে (চিত্র-১১) দেখানো হয়েছে ফর্মের ডিজাইনটি। ডানের চিত্রে (চিত্র-১২) তুলে ধরেছি রান করার পরে বোতামে ক্লিক করার ফলাফল। আপনি যখন মেসেজ বাক্সের 'Ok' বোতামে ক্লিক করবেন তখনই ফিল্ডে কিছু লিখার জন্য প্রোগ্রাম অপেক্ষায় থাকবে।



চিত্র-১১: (বায়ে) ফর্মের ডিজাইন। (ডানে) প্রোগ্রাম চলাকালে বোতামে ক্লিক করলে যে মেসেজ আসবে তার দৃশ্য।

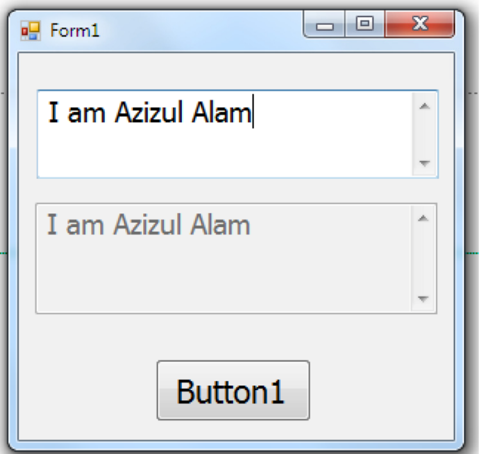
ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

আগের পৃষ্ঠায় বর্ণিত উদাহরণের দু'টি Sub নিচে তুলে ধরেছি। এরপর (চিত্র-১৩) প্রোগ্রাম চলাকালে প্রথম টেক্সট্ বাক্সের লেখা বাক্স থেকে মাউস সরে নিয়ে আসার পর দ্বিতীয় টেক্সট্ বাক্সে লেখা কপি হয়েছে দেখা যাচ্ছে।

```
Public Class Form1
   'This sub is the button click event code
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      MsgBox("Please type something in the first text box.") 'call messageBox to display
      'instruction
   End Sub

   'Catching MouseLeave event and doing something
   Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
      If TextBox1.Text = "" Then Exit Sub 'checks if there is text in the first field. if not program ends.
      TextBox2.Text = TextBox1.Text  'if there is text, this line executed, copies
      ' text into 2nd field.
   End Sub
End Class
```

উক্ত সাব দু'টো কিভাবে কাজ করে: প্রোগ্রাম রান করলেই পুরো ফর্মটি প্রদর্শিত হবে। আপনি যদি বোতামে ক্লিক করেন তাহলে প্রথম সাবটি কাজ শুরু করবে। এই সাবে আমরা নির্দেশ দিয়েছি: একটি মেসেজ বাক্স প্রদর্শন করতে যেখানে লিখা থাকবে, **"Please type something in the first text box."** কথাটি। **'Ok'** ক্লিক করার পর আগের অবস্থায় ফিরে যাবে। এবার আপনি যদি কিছু লিখেন এবং মাউসটি বাক্স থেকে বের করেন তাহলে সাথে সাথে লেখাটুকু এমনিতেই দ্বিতীয় বাক্সে চলে যাবে। এই কাজটি সম্ভব হয় ১ম টেক্সট্ বাক্সের মাউসলীভ সাব সক্রিয় হওয়ার কারণে। এ সাবে আমরা নির্দেশ দিয়েছি: প্রথমে পরীক্ষা করো, বাক্সে কিছু লেখা হলো কি না? যদি না হয়- সাব থেকে বের হও। যদি লেখা থাকে তাহলে দ্বিতীয় লাইনের নির্দেশ পালন করো। অর্থাৎ ১ম বাক্সের যাবতীয় লেখা ২য় বাক্সে লিখ। এরপর সাব থেকে বের হও।



চিত্র-১৩: প্রোগ্রাম Run করে প্রথম টেক্সট্ বাক্সের লিখা স্বয়ংক্রিয়ভাবে দ্বিতীয়টি লেখা হওয়ার দৃশ্য।

২য় উদাহরণ: Function

উপরের ১ম উদাহরণে আমরা Sub-এর ব্যবহার দেখেছি। এবার একটি নিজের তৈরি ফাংশন (Function) উপস্থাপন করছি। ভিজুয়্যাল বেসিকে Function সর্বদাই কোনো এক বা একাধিক ভ্যাল্যু (value) রিটার্ন করে। এটা হতে পারে প্রাইভেট বা পাবলিক বা আরো কোনো ধরনের। ইতোমধ্যে আপনি ফাংশনের সিন্টেক্স (Syntax) দেখেছেন। যাক, উক্ত প্রজেক্টে এখন আমরা একটি ফাংশন যোগ করবো।

** **মনে রাখবেন: ফাংশন ব্লক ফর্মের ক্লাসের ভেতর যে কোনো স্থানে লিখা যায়। এটাকে যে কোনো সাব বা অপর ফাংশন থেকে কল (call) করা হয়ে থাকে। ফাংশন (এবং সাবকে) বার বার কল করা সম্ভব। আর সর্বদাই ফাংশন থেকে ফলাফল (ভ্যালু) আসবে। ভ্যালুর ধরন (type) যে কোনোটি হতে পারে, যেমন: text, number, etc.** **

উদ্দেশ্য: আমাদের ফাংশনের কাজ হলো, আপনি মোট ক'টি অক্ষর টাইপ করেছেন তা একটি মেসেজ বাক্সে দেখানো। সুতরাং ফাংশনের রিটার্ন ভ্যালু হবে একটি ইন্টেজার (আস্ত) সংখ্যা। পরের পৃষ্ঠায় এ সম্পর্কে বিস্তারিত দেখুন।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

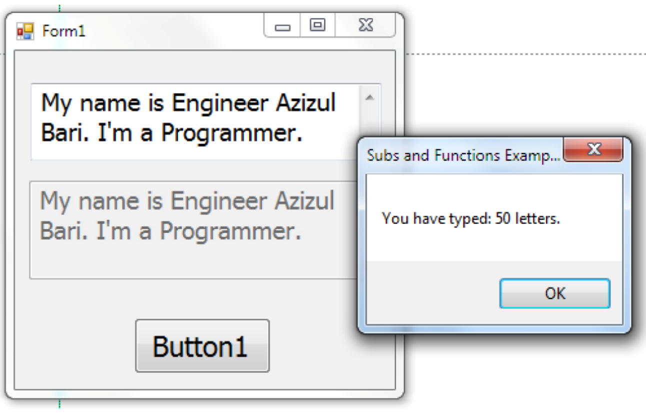
আমাদের ফর্ম ডিজাইনে কোনো পরিবর্তনের প্রয়োজন নেই। আমাদের কাজ হলো ফাংশন ব্লকটি ফর্মের ক্লাসের ভেতর লেখা। এরপর এটি অন্য কোনো কোড ব্লক থেকে কল করা। এই যা। নিচে ফাংশন কোডসহ পুরো কোড তুলে ধরেছি।

```
Public Class Form1
    'This sub is the button click event code
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("Please type something in the first text box.") 'call messageBox to display
        'instruction
    End Sub
    'Caching MouseLeave evet and doing something
    Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
        If TextBox1.Text = "" Then Exit Sub 'chacks if there is text in the first field
        TextBox2.Text = TextBox1.Text   'if there is text this line executed. copies
        ' text into 2nd field.
        'function call from message box
        MsgBox("You have typed: " & NumOfChar(TextBox1.Text) & " letters.")
    End Sub
    'function returns number of characters in TextBox1
    Function NumOfChar(Supply As String) As Integer
        If TextBox1.Text = "" Then 'in case there aren't any characters in the textBox1
     'although it is checked at the beginning of the sub. This If --- End If block is for extra solidarity.
            Return 0 'empty textBox means no character.
        End If
        Return Len(TextBox1.Text) 'counts character and returns integer
    End Function
End Class
```

প্রোগ্রামে ফাংশনের ব্যবহার: আমাদের সৃষ্ট NumOfChar (Supply) ফাংশন কিভাবে ব্যবহৃত হয়েছে, তা একটু লক্ষ্য করুন। টেক্সটবক্স১ এর 'TextBox1_MouseLeave' ইভেন্ট সাব থেকে এটি কল করেছি। ফলাফল একটি সংখ্যা যা মেসেজ বাক্সে লিখা হবে। অন্যকথায় মেসেজ বাক্সের ভেতর থেকেই আমরা কলটি করেছি। চাইলে এভাবেও হতো:

```
Dim NumOfLetters as Integer
NumOfLetters = NumOfChar(TextBox1.Text) 'Function call
MsgBox("You have typed: " & NumOfLetters & " letters.")
```

উক্ত কোড দ্বারা প্রথমে আমরা NumOfLetters নামক একটি ভেরিয়েবলে বাক্সের অক্ষর সংখ্যা ফাংশন কল করে সংরক্ষণ করেছি। এরপর এটা মেসেজ বাক্স দ্বারা প্রদর্শিত হয়েছে। তবে এ ক্ষেত্রে দু'টি অতিরিক্ত লাইন কোডিং হয়েছে। যাক, নিচে ফাংশনের কাজ (চিত্র-১৪) চিত্রিত হয়েছে।

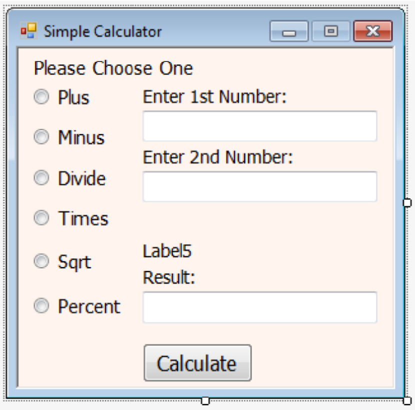


চিত্র-১৪: NumOfChar (Supply) ফাংশন -এর ব্যবহার ও ফলাফল।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

৩য় উদাহরণ: **Sub & Function**

এই উদাহরণে আমরা সত্যিকার অর্থে ব্যবহারযোগ্য একটি ক্যালকুলেটর তৈরি করতে যাচ্ছি। যে কোনো পিসিতে এটি ব্যবহার করা যাবে। প্রথমে আমরা ফর্মের ডিজাইন নিয়ে কিছু আলোচনা করবো। নিচের চিত্রে (চিত্র -১৫) এটি দেখানো হয়েছে। নিম্নে বর্ণিত পদ্ধতিতে প্রজেক্ট তৈরি করে ফর্ম ডিজাইন করুন।



চিত্র -১৫: সিম্পুল ক্যালকুলেটর প্রজেক্টের ফর্ম

1. একটি নতুন প্রজেক্ট তৈরি করুন। এর নামকরণ করুন SimpleCal
2. ফর্মের সাইজ প্রোপার্টি হবে: ৩১৫ x ৩১০। Name প্রোপার্টি হবে Simple Calculator। ফর্মের FormBorderStyle প্রোপার্টি হবে Fixed3D এবং MaximizeBox প্রেপার্টি হবে False।
3. ফর্মে ৫টি লেবেল থাকবে: label1 এর Text প্রোপার্টি হবে Please Choose One। label2 এর Text প্রোপার্টি হবে Enter 1st Number। label3 এর Text প্রোপার্টি হবে Enter 2nd Number। label4 এর Text প্রোপার্টি হবে Result:। label5 এর Text প্রোপার্টি হবে Label5- এটা দেওয়াই থাকবে।
4. ফর্মে থাকবে ৩টি টেক্সট্ বক্স: TextBox1, TextBox2 ও TextBox3। সবগুলোর Font Size প্রোপার্টি হবে 11। শুধু TextBox3 -এর ReadOnly প্রোপার্টি হবে True। আমরা চাইনা এই বাক্সে ব্যবহারকারী কিছু লিখবেন।
5. ফর্মে মোট ৬টি রেডিও বাটন থাকবে: এগুলোর ক্যাপশন নাম (Text property) হবে যথাক্রমে, Plus, Minus, Divide, Times, Sqrt এবং Percent। এদের (Name) প্রোপার্টি হবে যথাক্রমে, RDplus, RDminus, RDdivide, RDtimes, RDsqrt এবং RDpercent। এই প্রোপার্টি সঠিকভাবে থাকা চাই। এ নামগুলো কোডে ব্যবহৃত হবে।
6. সবশেষে ফর্মে একটি বোতাম থাকবে। এর Font Size প্রোপার্টি হবে 12 এবং Text প্রোপার্টি হবে Calculate। চাইলে ফর্মের BackColor প্রোপার্টি বদলাতে পারেন। আমি হালকা হলুদ (SeaShell) দিয়েছি।

ডিজাইন শেষ হওয়ার পর একবার Start মেনুতে ক্লিক করে রান করে দেখুন। লক্ষ করুন, ফর্ম বড় কিংবা ছোট করা যাবে না। আমরা এটাই চাই। ক্লোজ বাক্সে ক্লিক করে বন্ধ করুন।

কোডিং

এ উদহরণে মূল কোড লিখিত হবে Calculate বোতামের Click ইভেন্টে। তবে ফর্মের Load ইভেন্টে একটি ছোট্ট কোড লেখার প্রয়োজন। আমরা চাই রেডিও বোতামের Default একটি চয়েজ থাকবে। এটি হবে Plus নামক রেডিও বোতামটি (অর্থাৎ প্রথমটি)। সুতরাং ফর্মের লোড কোড হবে নিম্নরূপ:

```
' default value 'Plus' -radio button1 chosen
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        RDplus.Checked = True
    End Sub
```

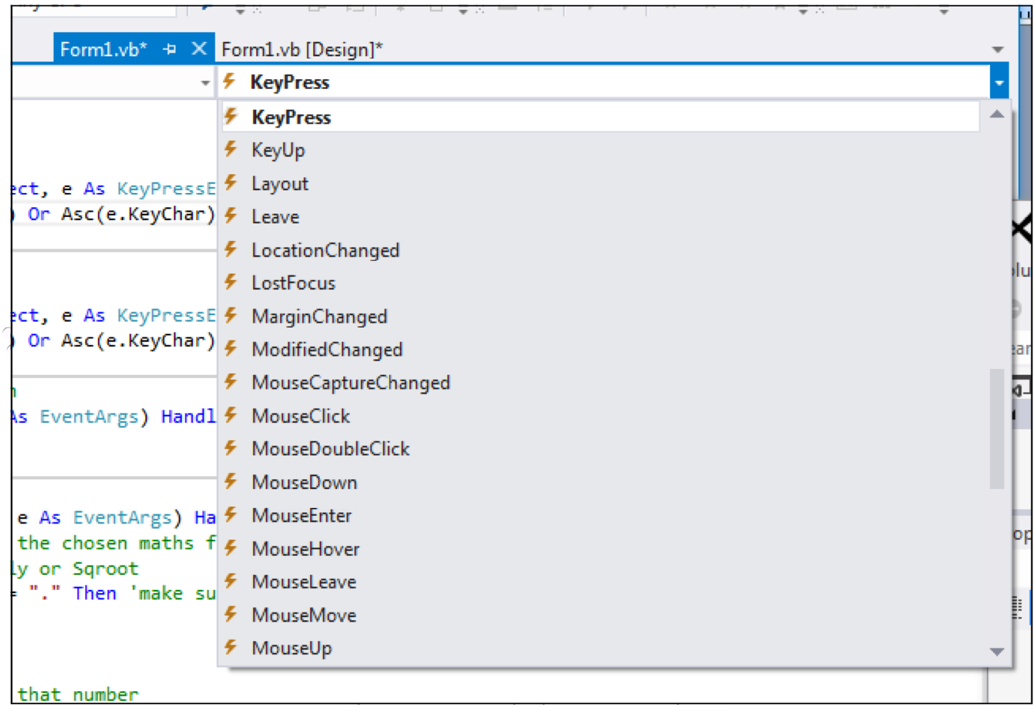
Form1.vb [design] ক্লিক করে ফর্মের যে কোনো স্থানে [কোনো বস্তুতে নয়] ডবল ক্লিক করলেই উক্ত সাবটি আসবে। এখানে শুধু মাঝের লাইনটি লিখবেন। এই লাইন দ্বারা বুঝানো হচ্ছে: ফর্মের Plus Radio button চেক করা থাকবে। এর দ্বারা এটাও বুঝাচ্ছে যে, ব্যবহারকারী উভয় টেক্সট্ বাক্সে সংখ্যা লিখে Calculate বোতামে ক্লিক করলে উভয়টির যোগফল আসবে।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

Calculate বোতামের Click ইভেন্টের কোডিংয়ে যাওয়ার পূর্বে আমরা আরো দু'টি ছোট্ট কোডিং করবো। এগুলো অনুরূপ এবং লিখিত হবে TextBox1 ও TextBox2 -এর KeyPress নামক ইভেন্টে। এই সাবটি পেতে হলে প্রথমে টেক্সটবাক্সে ডবল ক্লিক করুন। এতে টেক্সটবাক্সের TextChanged নামক ইভেন্টটি আসবে। এটা আমাদের প্রয়োজন নেই। তবে উপরে ডানদিকের মেনু থেকে চয়েজ করে নিতে হবে টেক্সটবাক্সের KeyPress নামক ইভেন্টটি (নিচরে চিত্রটি দেখুন)। নিচের সাবটি আসবে। এই সাবের মধ্যেই আমরা এক লাইন কোড লিখেছি।

চিত্র -১৬: কিভাবে টেক্সটবাক্সের KeyPress ইভেন্ট সাব বের করবেন?




```
'Make sure only numbers can be entered
Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox1.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar
= ".") And (sender.Text.IndexOf(".") = -1)))
End Sub
```

ঠিক একই কোড যাবে TextBox2 এর KeyPress ইভেন্ট সাবেও:

```
'Make sure only numbers can be entered
Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox2.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar
= ".") And (sender.Text.IndexOf(".") = -1)))
End Sub
```

উপরের কোড লাইনটি ভালো করে শিখে নিন। এই কোড দ্বারা যে কোনো টেক্সটবাক্সে শুধুমাত্র সংখ্যা (দশমিক একটিমাত্র বিন্দুসহ '.') এবং Delete ছাড়া অন্যকিছু যাতে কেউ লিখতে পারেন না তা নিশ্চিত করে। প্রয়োজনে কোডটি নোট বইয়ে লিখে রাখুন।

যাক এ পর্যন্ত আমরা যা করেছি তা দেখার জন্য একবার প্রোগ্রাম রান করে দেখুন। উভয় টেক্সটবাক্সে সংখ্যা লিখুন এবং সংখ্যা ছাড়া অন্য কিছু লিখার চেষ্টা করুন। এবার আমরা আসল কোড নিয়ে বিস্তারিত আলোচনা করবো।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

Calculate বোতামের কোড

আমরা ইতোমধ্যে ফর্মের লোড ইভেন্ট ও সংখ্যা লেখার কীপ্রেস ইভেন্ট কোড নিয়ে আলোচনা করেছি। এই কোডগুলো দ্বারা প্রথমত Plus রেডিও বোতাম চয়েজ করা অবস্থা ও টেক্সটবাক্স দু'টোয় শুধুমাত্র সংখ্যা লিখা সম্ভব- তা নিশ্চিত করেছি। এবার আমরা ক্যালকুলেট বোতামের কোড নিয়ে আলোচনায় যেতে পারি। নীচে প্রথমে বোতামের কোডগুলো তুলে ধরলাম। এরপর প্রত্যেক লাইনের ওপর ব্যাখ্যা দেবো।

```
 'calculation button
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim ActWant As String = WhichCal() 'get the chosen maths fuction value
    ' ActWant= Add, Minus, Division, Multiply or Sqroot
      If TextBox1.Text = "" Or TextBox1.Text = "." Then
     Exit Sub 'make sure there is 'number in the first fld
      End If
'first fld has a number
'check if the user wants to get sqrt of that number
  If TextBox1.Text <> "" And ActWant = "Sqroot" Then
        TextBox2.Text = "" ' Note: if second textBox also has number it is
     'ignored and wiped
        Label5.Text = "√" & TextBox1.Text
        TextBox3.Text = Math.Sqrt(Val(TextBox1.Text)) 'calculate square root, display
                      'and exit
        Exit Sub
  End If
        'textBox1 has a number and chosen radio button is not sqroot
        'check if textBot2 has a valid number
      If TextBox2.Text = "" Or TextBox2.Text = "." Then 'make sure there is number in
                      'the first fld
       Exit Sub
      End If
      'check if the user trying to divide by 0
      If Val(TextBox2.Text) = 0 And ActWant = "Division" Then
          TextBox3.Text = "Division by 0 not valid!"
          Exit Sub
      End If
      'both textBox have valid numbers- calculate and display results
      If TextBox1.Text <> "" Or TextBox2.Text <> "" Then 'check again
          If ActWant = "Add" Then
              Label5.Text = TextBox1.Text & " + " & TextBox2.Text 'display info
              TextBox3.Text = Val(TextBox1.Text) + Val(TextBox2.Text) 'add and show
              Exit Sub
          End If
          If ActWant = "Minus" Then
              Label5.Text = TextBox1.Text & " - " & TextBox2.Text 'display info
              TextBox3.Text = Val(TextBox1.Text) - Val(TextBox2.Text) 'minus and show
              Exit Sub
          End If
          If ActWant = "Division" Then
             Label5.Text = TextBox1.Text & " ÷ " & TextBox2.Text 'display info
             TextBox3.Text = Val(TextBox1.Text) / Val(TextBox2.Text) 'divide and show
             Exit Sub
          End If
```

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

```
        If ActWant = "Multiply" Then
           Label5.Text = TextBox1.Text & " x " & TextBox2.Text 'display info
           TextBox3.Text = Val(TextBox1.Text) * Val(TextBox2.Text) 'multiply and show
           Exit Sub
        End If
        If ActWant = "Percent" Then
           Label5.Text = TextBox2.Text & "% of " & TextBox1.Text 'display info
           TextBox3.Text = Val(TextBox1.Text) * Val(TextBox2.Text) / 100 'get percent
                                 'and show
           Exit Sub
        End If
     End If
End Sub
```

==================================== ======= ============================== =======================

উপরের কোডের ব্যাখ্য

প্রথম লাইন: `Dim ActWant As String = WhichCal()` দ্বারা আমরা `ActWant` নামক একটি ভেরিয়েবল্ ঘোষণা করছি। এটার মধ্যে একটি শব্দ থাকবে যা নির্ভর করবে ব্যবহারকারী কোন্ রেডিও বোতাম চয়েজ করেছেন তার ওপর। লক্ষ করুন একই লাইনে ভেরিয়েবলের ভ্যালু নির্ধারণ করা হয়েছে `WhichCal()` নামক ফাংশন কল করে। এই ফাংশন আমরা আলাদাভাবে ডিফাইন করেছি। এর ওপর আলোচনা একটু পরে আসছে। এখানে শুধু এটুকু জানতে হবে যে, এটা রিটার্ন করে ব্যাবহারকারী কোন্ রেডিও বোতাম চয়েজ করেছেন তার একটি নাম। সুতরাং প্রথম লাইন থেকে আমরা জানলাম ব্যবহারকারী কোন্ ক্যালকুলেশন করবেন। এবার দেখে নেই `WhichCal()` ফাংশনের কোডটি।

```
Function WhichCal() 'Returns values according to which radio button clicked
        If RDplus.Checked = True Then
            Return "Add"
        End If
        If RDminus.Checked = True Then
            Return "Minus"
        End If
        If RDdivide.Checked = True Then
            Return "Division"
        End If
        If RDtimes.Checked = True Then
            Return "Multiply"
        End If
        If RDsqrt.Checked = True Then
            Return "Sqroot"
        End If
        If RDpercent.Checked = True Then
            Return "Percent"
        End If
        Return "Add"
 End Function
```

লক্ষ করুন: এই ফাংশনের কোনো প্যারামিটার নেই। এটার কাজ হলো ব্যবহারকারী কোন্ রেডিও বোতাম ক্লিক করে রেখেছেন তা পরীক্ষা করে একটি নাম [অর্থাৎ, 'Add', 'Minus', 'Division', 'Multiply', 'Sqroot' কিংবা 'Percent'] ফেরৎ পাঠানো। ফাংশনের Return কিওয়ার্ড এ কাজটি করে। এই কিওয়ার্ড পরে ফাংশনের কাজ শেষ হয়ে যায়। পরের লাইনে আর যাবে না।

মনে করুন, আপনি চয়েজ করলেন Times নামক রেডিও বোতামটি- অর্থাৎ পূরণ অঙ্ক করবেন। এই ফাংশন একে একে চেক করে দেখবে কোন্ রেডিও বোতাম চেক করা অবস্থায় আছে। যেতেহু RDtimes নামক বোতামটি চয়েজ করা আছে- তাই এখানে এসে ফাংশন রিটার্ন করবে Multiply ভ্যালু। অন্যকথায়, এক্ষেত্রে ActWant এর মান হবে Multiply।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

ব্যবহারকারী কোন্ হিসাব করবেন তা প্রথমেই আমরা জেনে নিলাম কোডের প্রথম লাইন থেকে। এবার একটি চেক করা চাই। দেখতে হবে প্রথম বাক্সে সংখ্যা আছে কি না। এছাড়া শুধুমাত্র দশমিক পয়েন্ট '.' তো আর সংখ্যা নয়। নিচের `If - End If` ব্লকের কাজ এটি। সংখ্যা না থাকলে কিংবা শুধু '.' থাকলে সাব থেকে বেরিয়ে যাবে।

```
If TextBox1.Text = "" Or TextBox1.Text = "." Then 'make sure there is number in the first fld
   Exit Sub
End If
```

যদি গ্রহণযোগ্য সংখ্যা প্রথম টেক্সটবাক্সে থাকে তাহলে পরবর্তী ব্লকের কাজ শুরু হবে। এটা প্রথমে দেখে নেয় কোন্ অঙ্ক করা হবে- তারপর সিদ্ধান্ত নেয়। প্রথম টেক্সটবাক্সে সংখ্যা থাকলে ও ব্যবহারকারী এটির স্কোয়ার রুট বের করতে ইচ্ছুক হলে ব্লকের ভেতরের লাইনগুলো কাজ করে সাব থেকে বেরিয় যাবে। মনে রাখবেন- স্কোয়ার রুটের জন্য সংখ্যা লাগে মাত্র একটি। সুতরাং দ্বিতীয় বাক্সে কিছু আছে কি না তা চেক করা নিষ্প্রোজন।

```
If TextBox1.Text <> "" And ActWant = "Sqroot" Then
     TextBox2.Text = "" ' Note: if second textBox also has number it is ignored and wiped
     Label5.Text = "√" & TextBox1.Text
     TextBox3.Text = Math.Sqrt(Val(TextBox1.Text)) 'calculate square root display and exit
     Exit Sub
 End If
```

যদি প্রথম টেক্সটবাক্সে সংখ্যা থাকে এবং ব্যবহারকারী চান অন্য কোনো হিসাব করতে তাহলে দেখতে হবে দ্বিতীয় টেক্সটবাক্সে আরেকটি সংখ্যা আছে কি না। এই কাজটি করে পরের কোড ব্লকটি।

```
If TextBox2.Text = "" Or TextBox2.Text = "." Then 'make sure there is number in the second fld
    Exit Sub
End If
```

দ্বিতীয় টেক্সট বাক্সে গ্রহণযোগ্য সংখ্যা আছে তা নিশ্চিত হলেই পরবর্তী কোড ব্লক কাজ শুরু করবে। এটার কাজ হলো একটি বিশেষ কন্ডিশন চেক করা। আমরা জানি যে কোনো সংখ্যাকে ০ দ্বারা ভাগ করা যায় না। ব্যবহারকারী ঠিক এটি করছেন কি না তা চেক করে দেখে নিচের এই কোড ব্লকটি।

```
If Val(TextBox2.Text) = 0 And ActWant = "Division" Then
   TextBox3.Text = "Division by 0 not valid!"  'এই লাইনটি ডিসপ্লে ফিল্ডে প্রদর্শিত হবে যাতে ব্যবহারকারী বুঝতে পারেন।
   Exit Sub
End If
```

এ পর্যন্ত সকল চেক-টেক সঠিক মতো হয়ে গেলে আমরা নিশ্চিত হবো ব্যবহারকারী +, -, X, ÷ কিংবা % - এই ৫টির মধ্যে যে কোনো একটি হিসাব করবেন। আর কোন্টি করবেন তা সংরক্ষিত আছে ঐ প্রথম লাইন থেকে পাওয়া ActWant ভেরিয়েবলে। সুতরাং বাকি ৫টি ব্লকের কাজ হলে ActWant ভেরিয়েবলে সংরক্ষিত তথ্য মুতাবিক হিসাব করে প্রদর্শন করা। অন্য কথায় ভেরিয়েবলে যে নামটি আছে সে মুতাবিক ব্লক চয়েজ করে হিসাব করবে। আমরা এই সবগুলো ব্লক আর এখানে দেখাচ্ছি না- শুধু একটা দেখলেই বুঝতে সক্ষম হবো কি হচ্ছে। মনে করুন ভাগ হবে- তাহলে নিচের কোড ব্লকটি সক্রিয়া হয়ে প্রোগ্রাম শেষ করবে।

```
 If ActWant = "Division" Then
     Label5.Text = TextBox1.Text & " ÷ " & TextBox2.Text -এই লাইন দ্বারা তথ্য দেওয়া হচ্ছে কোন্ হিসাব হবে।
     TextBox3.Text = Val(TextBox1.Text) / Val(TextBox2.Text) 'divide and show
     Exit Sub
 End If
```

ব্লকের দ্বিতীয় লাইন দ্বারা আমার বুঝিয়েছি ব্যবহারকারী উভয় সংখ্যা দিয়ে কোন্ হিসাব করবেন। এখানে বলা হচ্ছে: লেবেল৫-এর টেক্সট হবে প্রথম সংখ্যা ভাগ দ্বিতীয় সংখ্যা। অনুরূপ অন্যান্য ক্ষেত্রেও এই তথ্য দেওয়া থাকবে। ব্যস! প্রোগ্রামের কোড নিয়ে আলোচনার ইতি হলো। ব্যাখ্যাগুলো প্রয়োজনে পুনরায় পাঠ করুন। কোডের একশন বুঝতে পারলে আপনি অনেকদূর এগিয়ে গেলেন।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

আমরা ইতোমধ্যে ভিজুয়্যাল বেসিক.নেট এর কয়েকটি উদাহরণ দেখেছি। এই পরিচ্ছেদেও আরো কিছু উদাহরণ দেখবো। তবে এসব উদাহরণের উদ্দেশ্য হবে ভাষার Variable Declaration এবং Data type সম্পর্কে অবগত হওয়া।

Variable কি?

যে কোনো প্রোগ্রামিং ভাষায় পরিবর্তনযোগ্য কিছু তথ্যের প্রয়োজন পড়ে। যে উপায়ে এটা সম্ভব তাহলো কম্পিউটারের মেমোরিতে তথ্যের ধরন চিহ্নিত করে বিশেষ নাম দ্বারা জায়গা সংরক্ষণ করা। আর প্রয়োজনে প্রোগ্রাম থেকে ঐ তথ্যটুকু এনে এর উপর কাজ, রদবদল, হিসাব-নিকাশ, পরিবর্তন ইত্যাদি করে পুনরায় সংরক্ষণ করা। এছাড়া দৃশ্যমান বস্তু যেমন টেক্সট বাক্স, লেবেল, লিস্ট বাক্স ইত্যাদিতে প্রদর্শন করা। মোটকথা ভেরিয়েবল ছাড়া কোনো কম্পিউটার প্রোগ্রাম লিখা প্রায় অসম্ভব। সুতরাং এটা কিভাবে ডিক্লায়ার করতে হয় এবং কোন্ উপায়ে কাজে লাগানো যায় তা জানা আমাদের জন্য একান্ত জরুরী।

কিভাবে ভেরিয়েবল ডিক্লারেশন করবেন?

প্রত্যেক ভেরিয়েবল নম্লিলিখিত উপায়ে ডিক্লায়ার করতে হবে:

১. স্কোপ বা ব্যবহারের ব্যপ্তি: ফর্মের কোথায় এটা ডিক্লায়ার করছেন তা-ই নির্ধারণ করে তার স্কোপ। ফর্মের ক্লাস ডিক্লারেশনের পরই (অর্থাৎ প্রথম লাইন বাদে) ডিক্লায়ার করলে ক্লাসের ভেতরস্থ সকল প্রসিডোর, সাব, ফাংশন ইত্যাদিতে এটা কাজে লাগানো যায়। অপরদিকে যে কোনো প্রসিডোর, সাব ও ফাংশনের ভেতর ডিক্লায়ার করলে শুধু ঐ সাবের ভেতরই এটা ব্যবহারযোগ্য হবে। নিচের উদাহরণ দু'টো দেখুন।

উদাহরণ ১. সর্বত্র ব্যবহারযোগ্য ভেরিয়েবল।

```
Public Class Form1
   Dim intNum As Integer 'intNum scope throughout rest of the program
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      intNum = 20 'Here intNum = 20
   End Sub
   Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
      intNum = 50 'Here intNum = 50
   End Sub
   Function NumOfChar()
      intNum = Len(TextBox1.Text) 'Here intNum = Number of characters in TextBox1
      Return intNum
   End Function
End Class
```

উদাহরণ ২. শুধুমাত্র সাব বা ফাংশনের ভেতর ব্যবহারযোগ্য।

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        dim EntText as String = TextBox1.Text 'only available in this sub
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        MsgBox(EntText) 'will not work -EntText is not declared error comes- because here it isn't useful.
    End Sub
End Class
```

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Data Type

কিম্পউটারে বুঝার জন্য যে কোনো তথ্যের ধরন বা টাইপ কি তা বলে দিতে হবে। ভেরিয়েবল ঘোষণার সময় এর Type-ও লিখে দিতে হয়। যেমন:

```
Dim MyVariable as String 'String is the type
Dim MyVariable as Integer 'Integer is the type
Dim MyVariable as Single 'Single (decimal number) is the type
```

এই গ্রন্থে ব্যবহৃত বিভিন্ন Type

1. Boolean - True or False
2. Char (single Character) - 0 through 65535 (unsinged)
3. Date - 0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999
4. Double (double precision floating point number)
5. Single (single precision floating point number)
6. Integer - -2,147,483,648 through 2,147,483,647 (signed)
7. Long (long Integer)
8. Object - Any type can be stored in a variable of type Object
9. Short (short integer) - -32,768 through 32,767 (signed)
10. String - 0 to approximately 2 billion Unicode characters

এবার আমরা উপরোক্ত প্রতিটি টাইপ ডিক্লারেশন ও ব্যবহার পদ্ধতির উদাহরণ দেখবো।

1. Boolean

```
Dim RadButOn As Boolean
' Check to see if RadioButton1 is ON or OFF.
If RadioButton1.Checked = True Then
        RadButOn = True
    Else
        RadButOn = False
End If
```

2. Char

```
'Normally we don't need this type. It is used to hold a single character only.
Dim MyChar as Char
MyChar = "A"
```

3. Date

```
 Dim date1 As Date = Date.Now
'gives local date and time: ex. 6/21/2016 1:34:14
 Dim date2 As Date = Date.UtcNow
'gives short date and coordinated universal time (GMT time without daylight change: ex. 6/21/2016 7:34:14)
 Dim date3 As Date = Date.Today
'gives just the short date: ex. 6/21/2016
```

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

4. Double

```
Dim DecOne as Double = 23.09898731
Dim DecTwo as Double = 5.98987654
MsgBox(DecOne * DecTwo) 'gives 138.360082185927
```

5. Single

```
Dim DecOne as Single = 23.09898731
Dim DecTwo as Single = 5.98987654
MsgBox(DecOne * DecTwo) 'gives 138.3601
```

উপরোক্ত Double এবং Single টাইপের মধ্যে পার্থক্য লক্ষ করুন। প্রথমটি দ্বিতীয়টি থেকে অধিক সঠিক।

6. Integer

```
Dim MyNum as Integer
```

লক্ষ করুন: ছোট আস্ত সংখ্যা নিয়ে কাজ করতে ইন্টেজার টাইপ ব্যবহার করা উত্তম।

7. Long

```
Dim MyNum as Long
```

লক্ষ করুন: বড় আস্ত সংখ্যা নিয়ে কাজ করতে লং টাইপ ব্যবহার করা উত্তম।

8. Object

```
'Any type can be stored in an Object Data Type
Dim v As Object
v = "17"
' v contains the 2-character String value "17".
v = v - 15
' v now contains the Integer value 2.
v = "H" & v
' v now contains the 2-character String value "H2"
```

9. Short

```
Dim ANum as Short 'Use short data type if you work with small integer to save memory
```

10. String

```
Dim j As String = "Joe said ""Hello"" to me."
Dim h As String = "Hello"
' The following messages all display the same thing:
' "Joe said "Hello" to me."
MsgBox(j)
MsgBox("Joe said " & """" & h & """" & " to me.")
MsgBox("Joe said """ & h & """ to me.")
```

সাধারণত আমরা Single, Long এবং Short Data type ব্যবহার করবো।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Array Variables

এ পর্যন্ত আমরা যেসব ভেরিয়েবল নিয়ে আলোচনা করেছি সেগুলোর প্রতিটিই মূলত একক কোনো ভ্যালু সংরক্ষণ করার ক্ষমতা রাখে। যেমন:

Dim A as integer 'A যে কোনো একটিমাত্র সংখ্যা সংরক্ষণ করতে পারে। তবে এটা পরিবর্তনশীল।
Dim Astring as string 'Asring যে কোনো একটিমাত্র string সংরক্ষণ করতে পারে। যেমন: Asring = "Ab"
মোটকথা, সাধারণ সকল ভেরিয়েবল যে কোনো একটি তথ্য সংরক্ষণ করতে পারে এবং তা পরিবর্তনশীল। তবে আপনি যদি চান ১০টি, ২০টি, ১০০টি কিংবা আরো বেশি সংখ্যা, নাম ইত্যাদি সংরক্ষণ করবেন তাহলে উপায় কি? প্রতিটি ভ্যালুর জন্য একেকটি নাম ডিক্লায়ার করতে পারেন। কিন্তু এভাবে ক'টি করবেন? শীঘ্রই আপনি প্রোগ্রাম করা থেকে ইস্তিফা দেবেন! ভাবার কোনো কারণ নেই। মাইক্রসফট ভিজুয়্যাল বেসিক আমাদেরকে শত শত বা এমনকি হাজার হাজার ভিন্ন সংখ্যা, নাম ইত্যাদি খুব সহজে ডিক্লায়ার করে সংরক্ষণের ব্যবস্থা করে দিয়েছে। এই ব্যবস্থার নামই হলো: Array Variables

Array variable declaration

এই ভেরিয়েবল ঘোষাণার নিয়ম এলো এরূপ:

```
Dim A(7) As Integer  'Here A is one-dimensional array with 8 elements.
```

লক্ষণীয় সকল অ্যারের উপদান ০ থেকে শুরু হয়। সুতরাং আপনি যদি আটটি ভিন্ন সংখ্যা বা অন্য কিছু ডিক্লায়ার করতে চান তাহলে ব্র্যাকেটের ভেতর ৭ লিখবেন। ১০টি চাইলে লিখবেন ৯, ১৫টি হলে লিখবেন ১৪। উপরের ডিক্লারেশনের ফলে যে ৮টি ইলেমেন্ট তৈরি হবে তাহলো নিম্নরূপ:

A(0), A(1), A(2), A(3), A(4), A(5), A(6), A(7)

কিভাবে ভ্যালু প্রবেশ করাবেন? কয়েকটি উপায় আছে। নিচে এগুলো উদাহরণ দ্বারা দেখিয়েছি।
১. মনে করুন আপনি ০ থেকে ৫ -এই ৬টি সংখ্যা অ্যারেতে সংরক্ষণ করবেন। আপনি এভাবে করবেন:

```
Dim MyNum (5) as Integer 'define array
Dim i as integer 'define a counter
For i = 0 to  5
    MyNum (i) = i
Next
```

এই প্রোগ্রাম অংশ রান করার পর MyNum নামক অ্যারে ভেরিয়েবলে থাকবে:
MyNum (0) = 0, MyNum (1) = 1, MyNum (2) = 2, MyNum (3) = 3, MyNum (4) =4 এবং MyNum (5) = 5

২. মনে করুন আপনি ব্যবহারকারীর কাছে ১০টি নাম লিখতে বলবেন এবং এগুলো একটি অ্যারে ভেরিয়েবলে সংরক্ষণ করবেন। আপনি তা এভাবে কতে পারেন:

```
Dim My10Names(9) As String ' define the array
Dim C As Integer ' define a counter
' fill the array
For C = 0 To 9
    My10Names(C) = InputBox("Enter the name please: " & C)
Next
```

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

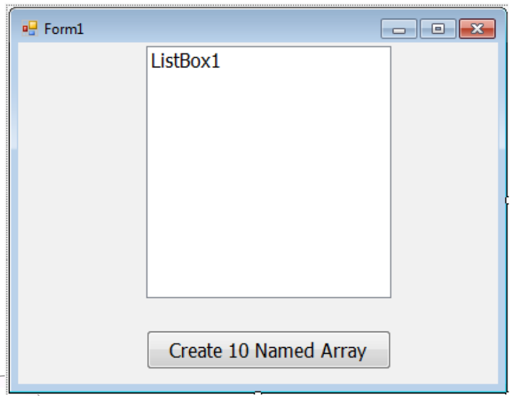
ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

সংরক্ষিত নামগুলো এক সঙ্গে দেখতে চাইলে আমরা ব্যবহার করতে পারি একটি লিস্টবক্স। নিম্নের কোড অংশ উপরের কোডের নিচে লিখলেই হবে।

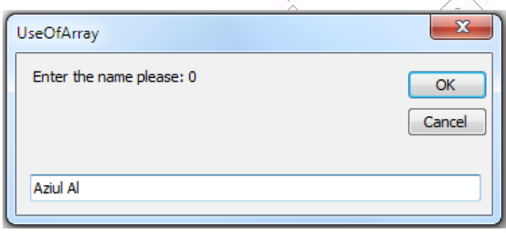
```
ListBox1.Items.Clear() ' display the result
For I = 0 To 7
    ListBox1.Items.Add(My10Names(I))
Next
```

Test 1: নতুন একটি প্রজেক্ট তৈরি করে এতে একটি লিস্টবক্স ও একটি বোতাম দিন। বোতামের নাম দিন 'Create 10 Names Array'। বোতামের ক্লিক ইভেন্টে উপরের কোড লিখে রান করুন। লিস্ট বাক্সে সব নাম এসছে কি? সমস্যা হলে নিচের ছবিগুলো দেখুন।

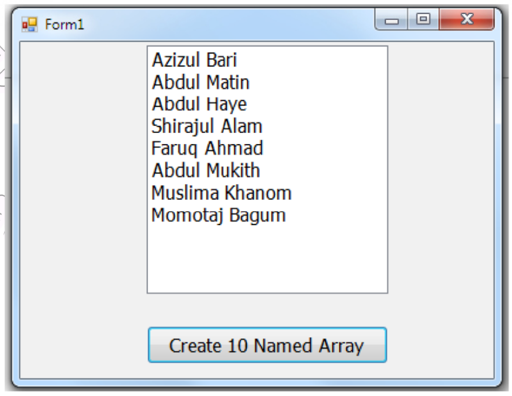
```
Public Class Form1
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
      Dim My10Names(9) As String 'define thearray
      Dim C As Integer ' define a counter
      ' fill the array
      For C = 0 To 9
         My10Names(C) = InputBox("Enter the name please: " & C)
      Next
      ListBox1.Items.Clear() ' display the result
      For I = 0 To 7
         ListBox1.Items.Add(My10Names(I))
      Next
   End Sub
End Class
```



চিত্র -১৭: ফর্ম ডিজাইন



চিত্র -১৮: নাম প্রবেশের ডায়লগ



চিত্র -১৯: রানের পর লিস্ট

মাইক্রসফট ভিজুয়্যাল বেসিক.কম উপরোক্ত 'Array' তৈরির সুযোগ দিয়ে অনেক তথ্য সংরক্ষণ ও ব্যবহারের ব্যবস্থা করে দিয়েছে। তবে এখানেই শেষ নয়। এ্যারে ছাড়াও আরো একটি গুরুত্বপূর্ণ তথ্য সংরক্ষণ ও ব্যবহার পদ্ধতি আবিষ্কার করেছে। এটা এ্যারে থেকেও উন্নতমানের এবং সহজ। এই ব্যবস্থার নাম collection। আমরা এখন এ সম্পর্কে বিস্তারিত আলোচনায় যাচ্ছি।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Collection Declaration & it's use -কালেকশন ঘোষণা ও এর ব্যবহার

Collection অনেকটা এ্যারের মতোই কাজ করে। তবে, কালেকশনের মধ্যে একাধিক মান বা ভেরিয়েবল্ সংরক্ষণ ও ব্যবহার করা যায় একই সঙ্গে। নিচে এ্যারে ও কালেকশের মধ্যে মূল ক'টি পার্থক্য তুলে ধরেছি।

ক. এ্যারের ইনডেক্স শুরু হয় ০ থেকে। কিন্তু কালেকশনের ইনডেক্স শুরু হয় ১ থেকে। যেমন:

Dim AnArray(3) - তে চারটি ইলেমেন্ট বা তথ্য থাকবে: AnArray(0), AnArray(1), AnArray(2) ও AnArray(3)। অপরদিকে:

```
Dim MyCollection as New Collection
MyCollection.add (1, "Sharif")
MyCollection.add (2, "Nasim")
MyCollection.add (3, "Abdullah")
MyCollection.add (4, "Khaled")
```

উপরের কালেকশনে চারটি ইলেমেন্ট সংরক্ষিত হবে। আমরা এদেরকে ব্যবহার করবো এভাবে রেফারেন্স করে: MyCollection(1), MyCollection(2), MyCollection(3) ও MyCollection(4)।

খ. আরেক পার্থক্য হলো ভেরিয়েবল টাইপ। এ্যারের মধ্যে সকল ইলেমেন্ট হতে হবে একই ধরনের। যেমন:

Dim MyArray (0 to 5) as integer -এখানে ০ থেকে ৫ পর্যন্ত ৬টি আস্ত সংখ্যা ঘোষণা করা হয়েছে।

Dim MyArray ("Aziz", "Matin", "Zakir") as string -এখানে তিনটি নাম যাদের সবার ভেরিয়েবল টাইপ হলো স্ট্রিং ঘোষণা করা হয়েছে। অপরদিকে কালেকশনের ইলেমেন্টগুলো বিভিন্ন ধরনের হতে পারে। যেমন:

```
Dim MyCollection as New Collection
MyCollection.add ("Sharif")
MyCollection.add (2)
MyCollection.add ("Hello!")
MyCollection.add ("01732562315")
```

গ. আরেক পার্থক্য হলো কালেকশনের ইলেমেন্ট সরাসরি যুক্ত ও সরানো যায়। অপরদিকে এ্যারে ফিক্স। যেমনে তেমনে ভ্যালু পরিবর্তন করা যায় না।

ঘ. আরো একটি পার্থক্য হলো, এ্যারে ইলেমেন্ট সহজেই পরিবর্তন বা অভাররাইট করা যায়। যেমন:

```
Dim MyArray (2)
MyArray (0) = 5
MyArray (0) = 7
```

কালেকশনের সময় এটা করা যাবে না। তবে অন্যভাবে করার ব্যবস্থা আছে। পরে আমরা সে সম্পর্কে আলোচনা করবো।

কালেকশনের কয়েকটি মেথড আছে যেমন:

```
Dim C As New Collection 'Declaration
C.Add (1, "Aziz") 'adding element
C.Add (2, "Hello!")
C.Remove(1) 'remove element in index 1
I = C.Count 'get number of elements in C as integer
```

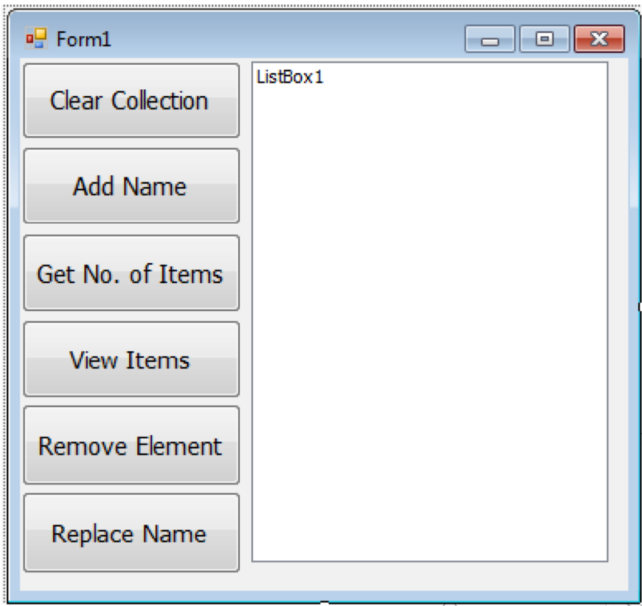
এখন আমরা একটি সাধারণ প্রোগ্রাম করবো। এতে কালেকশনের ব্যবহার সুস্পষ্ট হবে।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : ***Variable & Data Type***

CollectionDemo Application

১. প্রথমে CollectionDemo নামকরণে একটি ফর্ম প্রজেক্ট তৈরি করে নিচের চিত্রের মতো ডিজাইন করুন। এতে থাকবে ৬টি বোতাম ও ১টি লিস্ট বাক্স। দেখুন নিচের চিত্রটি (চিত্র: ২০)।



চিত্র -২০: CollectionDemo প্রজেক্টের ফর্ম

লক্ষ করুন, প্রতি বোতামের নামকরণ হবে দেখানো নামগুলোর মতো। লিস্ট বাক্সের নাম যা আছে তাই থাকবে। এবার নিচের কোডগুলো বর্ণিত বোতামের ক্লিক ইভেন্টে লিখুন।

১. Collection Declaration: এটা হবে ফর্মের ক্লাস ডিক্লারেশনের পরে:

```
Public Class Form1
    Dim MyCollection As New Collection
End Class
```

২. যথাক্রমে ‘Clear Collection’, থেকে ‘Replace Name’ বোতামের ক্লিক ইভেন্ট কোডসমূহ:

```
Public Class Form1
    Dim MyCollection As New Collection
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        ' this method clears all the elements in the collection
        MyCollection.Clear()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        ' read a name
        Dim Name As String
        Name = InputBox("Enter a Name") 'ask user to enter a name
        ' add the name into the list
        MyCollection.Add(Name)
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        'Show number of items in a message box.
        MsgBox("The number of items is: " & MyCollection.Count)
    End Sub
```

ভিজুয়াল বেসিক.নেট : *Variable & Data Type*

CollectionDemo Application

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        ' clear old content
        ListBox1.Items.Clear()
        ' insert the items into the list box
        Dim I As Integer
        For I = 1 To MyCollection.Count
            ListBox1.Items.Add(MyCollection(I))
        Next
End Sub

Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
        ' get element position
        Dim I As Integer
        I = InputBox("Enter the element number you want to remove:") 'ask user to enter
element number
        If I <= MyCollection.Count And I >= 1 Then 'check in the element number is valid
            MyCollection.Remove(I)
        Else
            'if invalid number enrered then show a message, reminding of number of elements
            MsgBox("You have 1 to " & MyCollection.Count & " items in the collection.")
        End If
 End Sub

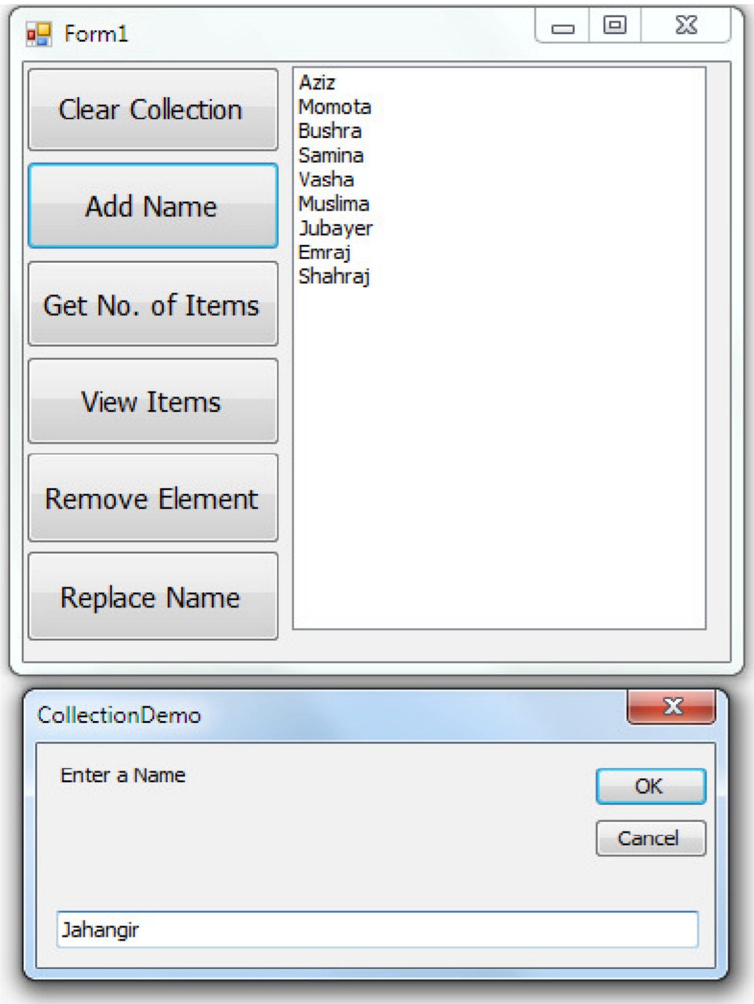
 Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click
        ' get element position
        Dim I As Integer
        Dim N As String
        I = InputBox("Enter the element number:") 'ask user to enter element number
        If I <= MyCollection.Count And I >= 1 Then 'check in the element number is valid
            N = InputBox("Enter the new name:") 'ask user to enter new name
            MyCollection.Add(N, , , I) 'add name in the element number
            MyCollection.Remove(I) 'remove old element
        Else
            'if invalid number enrered then show a message, reminding of number of elements
            MsgBox("You have 1 to " & MyCollection.Count & " items in the collection.")
        End If
    End Sub
End Class
```

উপরের কোড দ্বারা আমরা কালেকশন ডিক্লায়ার করা, ক্লিয়ার করা, নতুন ইলেমেন্ট যোগ দেওয়া, ক'টি ইলেমেন্ট আছে তা জানা, সবগুলো প্রদর্শন করা, মুছে দেওয়া এবং পরিবর্তন করার পদ্ধতি দেখিয়েছি। পরের পৃষ্ঠায় প্রোগ্রাম রানিং অবস্থার একটি চিত্র (চিত্র : ২১) দিয়েছি। একটু দেখে নিন।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

CollectionDemo Application



চিত্র -২১: CollectionDemo Application চলাকালীন দৃশ্য।

আমি আশাবাদী আলোচিত প্রোগ্রামটি কিভাবে কাজ করে তা আপনি সম্পূর্ণরূপে বুঝতে সক্ষম হয়েছেন। তবে, কোনো সমস্যা হলো পুরো পরিচ্ছেদটি পুনরায় পাঠ করুন। সকল কোডে ইংরেজিতে কমেন্ট দেওয়া আছে। এগুলো পাঠ করলে বুঝতে অনেকটা সহজ হবে।

এখন আমরা প্রাথমিক চারটে পরিচ্ছেদে কি শিখলাম তার সারসংক্ষেপ তুলে ধরছি।

১. পরিচ্ছেদ ১ তথা ভূমিকা থেকে আমরা জেনেছি ভিজুয়্যাল বেসিক.নেট ভাষার মাধ্যমে কি কি করা যায় এবং এ ভাষার মৌলিক গুণাগুণ ইত্যাদি।

২. দ্বিতীয় পরিচ্ছেদে আমরা শিখেছি:

ক. আই.ডি.ই সম্পর্কে।

খ. বিসমিল্লাহ প্রোগ্রাম।

৩. তৃতীয় পরিচ্ছেদে আমরা শিখেছি ভিজুয়্যাল বেসিক.নেট এর প্রোগ্রামিং কাঠামোর মৌলিক দু'টি বিষয়: Subroutine (Sub) এবং Function কিভাবে কাজ করে। আমরা উভয়টির কোডিং নিয়ম বা Syntax দেখেছি। প্রাকটিস প্রোগ্রাম নিয়ে আলোচনাও হয়েছে। আমরা Simple Calculator Application বানিয়েছি। এটা ছিলো একটি ব্যবহারযোগ্য প্রোগ্রাম। এ প্রোগ্রাম দ্বারা ৬টি পাটিগণিতের হিসাব করা যায়। এই প্রজেক্টের ডিজাইন ও কোডের ওপর বিস্তারিত ব্যাখ্যাও দেওয়া হয়েছে। আমি আশা করি আপনি এই প্রজেক্ট পাঠ করে কোডিং সম্পর্কে বেশ অভিজ্ঞ হয়ে ওঠেছেন।

৪. সবশেষে বর্তমান এই পরিচ্ছেদে আমরা ভিজুয়্যাল বেসিক.কম -এ ব্যবহারযোগ্য ভেরিয়েবল নিয়ে বিস্তারিত আলোচনা হয়েছে। আমরা দু'টি গুরুত্বপূর্ণ ভেরিয়েবল সম্পর্কেও উদাহরণসহ তথ্যাদি তুলে ধরেছি: ১. Array এবং ২. Collection

পরের পরিচ্ছেদে Looping নিয়ে আরো কিছু আলোচনা সংক্ষিপ্তভাবে করবো। এরপর আমরা ভিজুয়্যাল বেসিক.নেট প্রোগ্রামে সচরাচর ব্যবহৃত একটি ব্যবহারকারী দ্বারা তৈরি-করা ভেরিয়েবল নিয়ে আলোচনা করবো উদাহরণসহ। এই ভেরিয়েবলের নাম Structure।

৫ম পরিচ্ছেদ: *Looping in Program*

বিভিন্ন লুপিং : *Various Looping*

আমরা ইতোমধ্যে ভিজুয়্যাল বেসিক.নেট এর কোডিংয়ে For .... Next লুপিং এর ব্যবহার দেখেছি। এটার আরেকটি উদাহরণ হলো:

```
Dim I As Integer
For I = 1 To 20
    ListBox1.Items.Add(I)
Next
```

এই ফর-নেক্সট লুপ ১ থেকে ২০ পর্যন্ত সংখ্যা লিস্ট বাক্স ১-এ প্রদর্শন করবে। প্রত্যেকবার লুপিংয়ের সময় I-এর মাত্রা ১ বাড়তে থাকে। মাঝের লাইন এই আই I-এর মাত্রা লিস্ট বাক্সে লিখে দেয়। I-এর মান ২০ হওয়ার পর আরেকবার লুপিংয়ের জন্য ফিরে আসলে I-এর মান হবে ২১। সুতরাং লুপিং শেষ হবে, কারণ নির্দেশ দেওয়া আছে I হবে ১ থেকে ২০ -এর বেশিও না কমও না। এবার আমরা আরেক ধরনের লুপিং নিয়ে আলোচনা করবো।

Do Loop

এই লুপের নিয়ম নিম্নরূপ:

```
Do
   statement .....
   statement....
Loop While condition
```

অথবা

```
Do
   statement.....
   statement....
Loop Until condition
```

অথবা

```
Do While condition
   statement...
   statement...
Loop
```

অথবা

```
Do Until Condition
   statement...
   statement...
Loop
```

লক্ষ করুন: কন্ডিশন কোথায় আছে তা-ই নির্ধারণ করে কখন চেক করা বা বের হওয়া কিংবা থাকা নির্দিষ্ট বৈধ হবে। যেমন:

৫ম পরিচ্ছেদ: *Looping in Program*

বিভিন্ন লুপিং : *Various Looping*

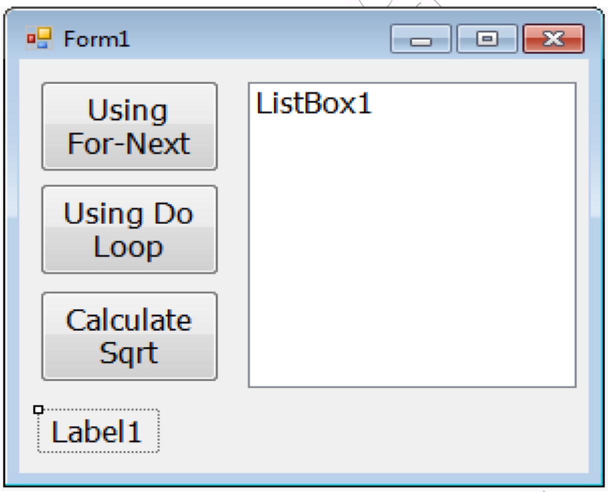
```
Do While L<10
    statement ...
    statement ...
Loop
```

এখানে প্রত্যেকবার চেক হবে লুপের কাজ শুরুর পূর্বে। এই লুপ কখনো শুরু না হওয়ার সম্ভাবনা আছে।

```
Do
    statement ...
    statement ...
Loop While L<10
```

এখানে চেক হবে লুপের কাজ শুরুর পরে। এই লুপ অন্তত একবার শুরু হবে।

এবার একটি উদাহরণ দেখে নিই। নিচের ফর্মের মতো একটি ফর্ম ডিজাইন করুন। এতে থাকবে তিনটি বোতাম, একটি লিস্ট বাক্স ও একটি লেবেল।



চিত্র -২২: লুপ উদাহরণ ফর্ম ডিজাইন

প্রথম বোতামের কোড হবে:

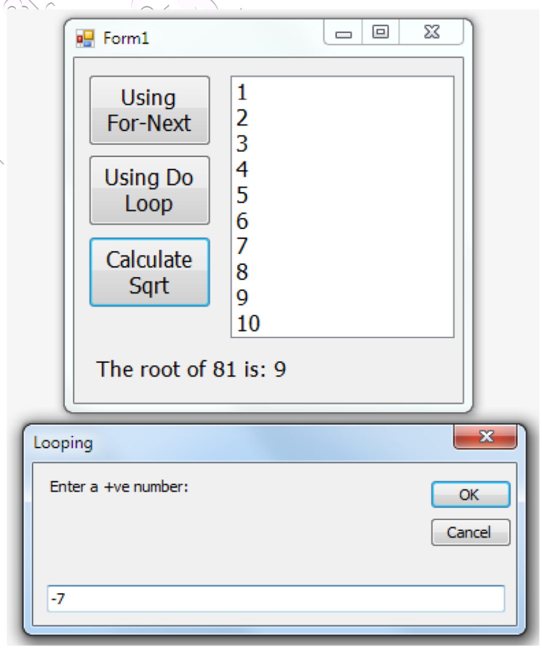
```
Dim I As Integer 'declare an integer variable
 ListBox1.Items.Clear() 'empty out list box
 For I = 1 To 10
    ListBox1.Items.Add(I) '1 to 10 into list box
Next
```

দ্বিতীয় বোতামের কোড হবে:

```
 Dim I As Integer 'declare integer variable
 I = 1 'initialize it
 ListBox1.Items.Clear() 'clear items in the list box
  Do
     ListBox1.Items.Add(I) 'add 1 to 10 in the list box
           I = I + 1 'add 1 to I
  Loop While (I < 11) 'check. I must be 11 to exit the loop
```

তৃতীয় বোতামের কোড হবে:

```
Dim I As Double 'declare double type variable (decimal number).
Do
I = InputBox("Enter a +ve number:") 'ask for a positive number.
'calculate sqrt and display in the label
    If I >= 0 Then Label1.Text = "The root of " & I  & " is: " &
Math.Sqrt(I)
Loop
```



চিত্র -২৩: লুপ উদাহরণ রানটাইম

লক্ষ করুন: প্রথম দু'টি বোতামের লুপ কোড এমনিতেই শেষ হবে। কিন্তু তৃতীয় বোতামের লুপ কোড ব্যবহারকারী নেগেটিভ সংখ্যা দ্বারা শেষ করবেন। অন্যথায় এই লুপ চিরকাল চলতে থাকবে! একে বলে নেভার এন্ডিং লুপ [Never ending loop]। প্রোগ্রামারকে এ ব্যাপারে সতর্ক থাকা চাই। অনাকাঙ্ক্ষিত নেভার এন্ডিং লুপে চলে গেলে কম্পিউটার ক্রাশ করতে পারে। যেমন আমরা যদি লিখে ফেলি (এটা করবেন না!):

```
dim a as integer =1
For b = 1 to a
    a = a+1
Next b
```

এখানে মাঝের লাইন দ্বারা a-এর মাত্রা এক এক করে বাড়তে থাকবে। সুতরাং b-এর মাত্রা কখনও শেষ হবে না এবং লুপও ক্লোজ হবে না। কম্পিউটার ক্রাশ করবে। উপরের লুপে `If I >= 0 Then` না থাকলে প্রোগ্রাম বন্ধ করা যেতো না।

৬ষ্ঠ পরিচ্ছেদ: *Structure*

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

একাধিক ভিন্ন ভেরিয়েবল্ নিয়ে একেকটি Structure তৈরি হয়। এটা মূলত প্রোগ্রামার দ্বারা বিভিন্ন ভেরিয়েবল একটি ইউনিট হিসাবে সৃষ্টি করার সুযোগ যা অনেকটা সুবিধাসম্পন্ন। উদাহরণ দ্বারা বিষয়টি পরিষ্কার হবে।
মনে করুন আপনি কোনো কোম্পানির কর্মচারীদের বিভিন্ন তথ্য একত্রে সংরক্ষণ করতে চান। তথ্যাদি হবে নাম, ঠিকানা, টেলিফোন নাম্বার, বয়স, বেতন, কাজের নাম ইত্যাদি। ইতোমধ্যে আমরা যা জেনেছি তার দ্বারা এটাই নির্দেশ করে যে প্রতিটি ভিন্ন তথ্য একেকটি আলাদা ভেরিয়বলে সংরক্ষণ করতে হবে। যেমন:

Dim EName as String, Dim EAddress as String, Dim ETelephone as String, Dim EAge as Integer, Dim ESalary as Decimal, Dim EJobTitle as String

এখন অপর আরেক কর্মচারীর জন্য অনুরূপভাবে ভেরিয়েবল ডিক্লারেশন দরকার। অন্য আরেক জনের জন্যও এরূপ করতে হবে। সুতরাং দশজন কর্মচারী হলে ৬০ টি ভেরিয়েবল লাগবে। বেশি হলে আরো বেশি লাগবে। এভাবে প্রোগ্রামিং করলে আপনি শীঘ্রই ইস্তিফা দেবেন! আসলে এরূপ করার দরকার নেই। আপনি Structure ব্যবহার করে একবার মাত্র ভেরিয়েবল ডিক্লারেশ করে কাজ সারতে পারেন। কিভাবে? এই দেখুন:

Structure ডিক্লারেশন

বিভিন্ন প্রোপার্টি একত্রে একটি Structure-এ এভাবে ডিক্লারেশন করা যায়:

```
Structure EmployeeInfo
    Dim EName as String
    Dim EAddress as String
    Dim ETelephone as String
    Dim EAge as Integer
    Dim ESalary as Decimal 'decimal number data type
    Dim EJobTitle as String
End Structure
```

এখন আমরা নতুন একটি ডাটা টাইপ তৈরি করলাম যার নাম EmployeeInfo। এর ভেতর আছে ৬টি ভিন্ন তথ্য। এবার EmployeeInfo-কে আমরা ডাটা টাইপ হিসাবে ডিক্লায়ার করতে পারি যেমনটি পারি String, Integer ইত্যাদি। সুতরাং আমরা ডিক্লায়ার করবো এভাবে:

```
Dim A as EmployeeInfo
A.EName = "Muhammad Abdus Samad"
A. EAddreaa ="10 Bashundara R/A, Maulvi Bazar, Bangladesh"
A.ETelephone ="01235 215226"
A.EAge = 35
A.ESalary = 32500.00
A.EJobTitle = "Managing Director"
```

স্টোর করা তথ্য আনতে হবে এভাবে: A.ETelephone, A.ESalary = 32500.00 ইত্যাদি। এ হলো কর্মচারী A। আমরা একই ভেরিয়েব টাইপ ব্যবহার করে আরো কর্মচারীর তথ্য সংরক্ষণ করতে পারি। যেমন:

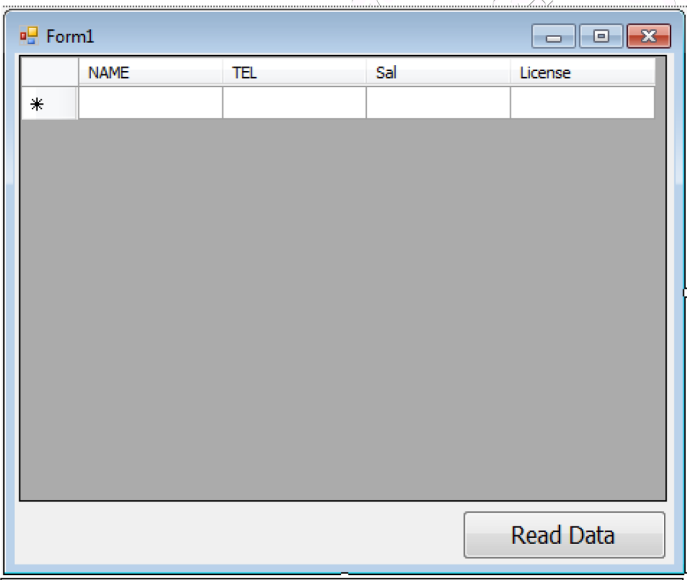
৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

```
Dim B as EmployeeInfo
B.EName = "Abdur Rahim"
B. EAddreaa ="99 Sunamgonj Rd, Patantula R/A, Sylhet, Bangladesh"
B.ETelephone ="01723658211"
B.EAge = 33
B.ESalary = 22500.00
B.EJobTitle = "Chief Supervisor"
```

অনুরূপ আমরা যতোজন কর্মচারীর প্রয়োজন ততোজনের তথ্য EmployeeInfo এর মধ্যে সংরক্ষণ করতে পারি। সুতরাং বুঝাই যাচ্ছে Structure একটি গুরুত্বপূর্ণ ডিক্লারেশন।
এবার আমরা Structure ব্যবহারের একটি উদাহরণ দেখাবো। নিচের পদক্ষেপগুলো অনুসরণ করুন।

১. নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে যোগ করুন ১টি বোতাম ও একটি ডাটাগ্রিডভিউ [DataGridView]। গ্রিডে থাকবে: Name, Tel, Sal, License -এ চারটি কলাম। দেখুন নিচের চিত্রটি।



চিত্র -২৪: স্ট্রাকচার প্রজেক্টের ফর্ম

২. ফর্মের যে কোনো স্থানে ডবল ক্লিক করুন। কোড জানালা আসবে। এতে লিখুন নিচের কোডটি:

```
Public Class Form1
    'declare the structure
    Structure PersonInfo
        'person name
        Dim Name As String  'person telephone number
        Dim Tel As String
     'person salary
        Dim Sal As Decimal
     'person license number
        Dim LincenseNumber As String
    End Structure
' this is the array to store persons' info
   Dim Info() As PersonInfo
End Class
```

এ কোড ব্লক দ্বারা আমরা PersonInfo নামে একটি নতুন ভেরিয়েবল ডিক্লায়ার করেছি। এ ব্লকটি ফর্মের ক্লাস ডিক্লারেশন পরেই থাকবে। আলাদা কোনো সাবরুটিনের ভেতর একে ডিফাইন করা যাবে না। PersonInfo হলো একটি Stucture ভেরিয়েবল। এ ভেরিয়েবলের ভেতর কয়েকটি ভিন্ন ভেরিয়েবল ডিক্লারেশন আছে। এ ক্ষেত্রে এগুলো হলো: Name, Tel, Sal এবং LicenseNumber। লক্ষ করুন, এসব ভেরিয়বলের তিনটি String Type ও একটি Decimal Type ভেরিয়বল ডিক্লারেশ করা হয়েছে। স্ট্রাকচারের মধ্যে এভাবে ভেরিয়েবল ডিক্লারেশনের ফলে এটা সম্ভব হয়েছে। প্রোগ্রামের সর্বত্র পাওয়ার নিশ্চয়তার জন্য Info() নামক একটি PersonInfo টাইপ এ্যারে ভেরিয়েবলও এখানে ডিক্লায়ার করেছি। এই এ্যারের মধ্যে ব্যক্তির তথ্যাদি সংরক্ষণ করা হবে। এখন আমাদেরকে PersonInfo নামক ভেরিয়েবলে তথ্য ঢুকাতে হবে। এজন্য আমরা Read Data বোতামটি ব্যবহার করবো। এর ক্লিক ইভেন্ট হ্যান্ডলারে আমরা পরের পৃষ্ঠার কোডটি লিখবো:

৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

```
 ' read the information and store it in an array
  Public Sub ReadInfo()
     Dim I As Integer 'I for For-Next Loop
     Dim N As Integer 'N for number of persons
     On Error GoTo ChkNum ' incase the user enters a wrong entry [not number]
     N = InputBox("Enter the number of people") 'first enter how many person
     ReDim Info(0 To N - 1) 'redimension the array to 0 to number of person
     ' read info here
     For I = 0 To N - 1
        Info(I).Name = InputBox("Enter the name of person")
        Info(I).Tel = InputBox("Enter the telephone number")
        Info(I).Sal = InputBox("Enter the salary")
        Info(I).LincenseNumber = InputBox("Enter license number")
     Next
     Exit Sub
ChkNum: 'label jump
     Exit Sub
  End Sub
```

উক্ত সাবটি কিভাবে কাজ করে তা আপনার মোটামুটি জানা হয়ে গেছে আশারাখি। এরপরও আবার বুঝিয়ে দিচ্ছি। প্রথমত সাবরুটিনটি নিজের বানানো মনে রাখবেন। আপনাকে সবকিছু নিজেই টাইপ করতে হবে। প্রথম লাইন দ্বারা ইন্টেজার ভেরিয়েবল ডিক্লারেশ করা হয়েছে। এই ইন্টেজার ফর-নেস্ট লুপের গণক হিসাবে ব্যবহৃত হয়েছে। দ্বিতীয় লাইনে ঘোষিত ইন্টেজার হলো লোকসংখ্যা। এটার মাত্রা চতুর্থ লাইনে নিশ্চিত হবে। ব্যবহারকারী যতো ঢুবাকেন তা-ই হবে এর মাত্রা। তৃতীয় লাইনটিকে বলে Error Handling। ভুলবশত ব্যবহারকারী সংখ্যা না ঢুকিয়ে অন্য কিছু ঢুকালে এই লাইন নিশ্চিত করবে সাব থেকে বের হতে। অন্যথায় প্রোগ্রাম ক্রাশ করবে। এই লাইন লেভেলে জাম্প করবে।

এবার সংগ্রহিত তথ্যাদি ডাটা-গ্রিড ভিউ-এ প্রদর্শনের প্রয়োজন। এজন্য আমরা নিজে একটি সাবরুটিন বানিয়েছি। এ'টি ফাংশনের মতো কাজ করবে। যখনই আমরা একে কল করবো তখনই এটা সক্রিয় হবে।

```
 ' fill the data grid with array info
  Public Sub FillDGV(ByVal DAT() As PersonInfo, ByVal DGV As DataGridView)
     DGV.Rows.Clear()
     Dim I As Integer
     For I = 0 To DAT.Length - 1
        On Error GoTo endSub 'error handling. Jump to endSub label and exit
        DGV.Rows.Add(DAT(I).Name, DAT(I).Tel, DAT(I).Sal, DAT(I).LincenseNumber)
     Next
     Exit Sub
endSub: 'label jump
     MsgBox("Enter a whole number please!")
     Exit Sub
  End Sub
```

উপরের সাব কিভাবে কাজ করে তা একটু তলিয়ে দেখা যাক। প্রথমত জেনে রাখি, এই সাবটি কল করা হবে নিচের দু'টি বোতামের সাব থেকে। কল করলেই এটা কাজ শুরু করবে। প্রথম লাইন তথা DGV.Rows.Clear() সকল রো ক্লিয়ার করবে। দ্বিতীয় লাইনে আমরা ফর-নেস্ট লুপের জন্য একটি কন্ট্রোল ইন্টেজার ডিক্লায়ার করেছি। লক্ষ করুন: ক'বার ফর-নেস্ট লুপ ঘুরবে তা নির্ভর করে ঠিক ক'জন ব্যক্তির তথ্য আমাদের এ্যারের মধ্যে সংরক্ষিত আছে- তার ওপর। লুপের লাইনটিতে DAT(I).Name, DAT(I).Tel, DAT(I).Sal, DAT(I).LincenseNumber দ্বারা একে একে সবগুলো রো পূর্ণ হবে। এবার দেখে নিই বাকি কোডগুলো।

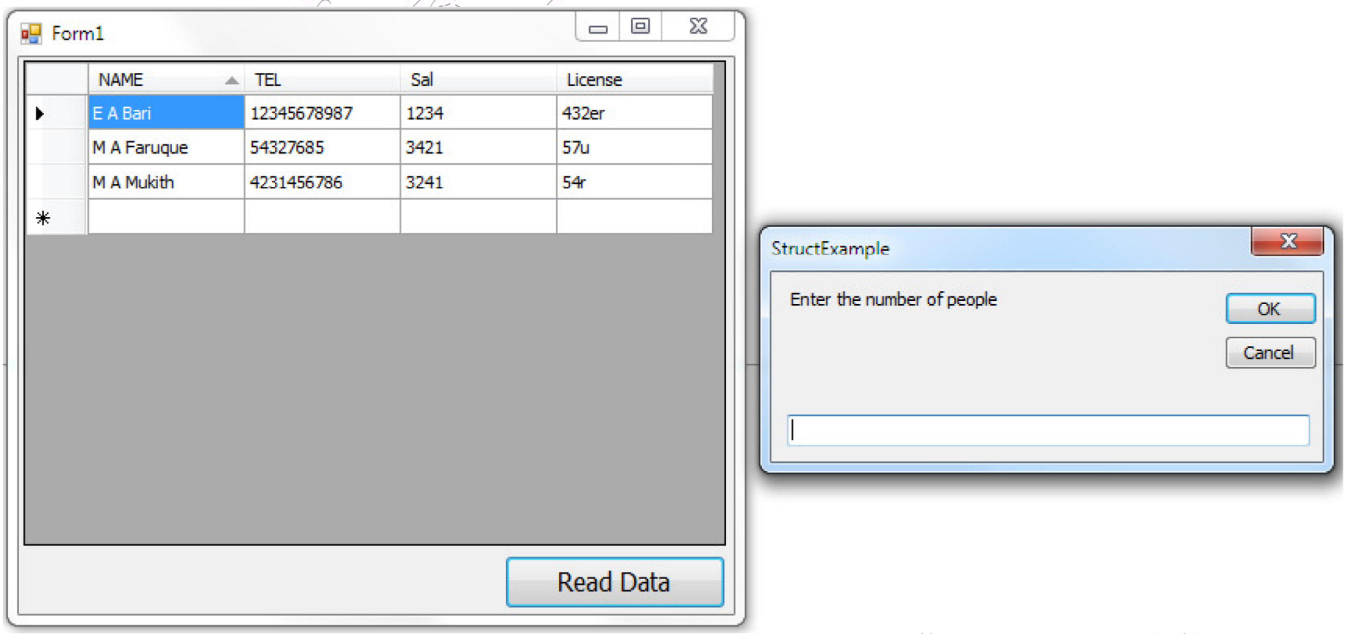
৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

Read Data বোতাম দ্বারা আমরা ব্যক্তিদের তথ্য সংগ্রহ করবো। এই বোতামের ক্লিক ইভেন্টে নিচের কোডটি লিখতে হবে। এই বোতাম দ্বারা ব্যবহারকারী প্রয়োজন পরিমাণ তথ্য ঢুকানোর পর গ্রিডে সবকিছু প্রদর্শিত হবে।

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
	ReadInfo() 'call sub to enter data by user
	FillDGV(Info, DataGridView1) 'call sub to display data in the grid
End Sub
```

লক্ষ করুণ এই সাবটির মধ্যে মাত্র দু'টি লাইন আছে। উভয় লাইন দ্বারাই আমরা ইতোমধ্যে ব্যাখ্যাত নিজের তৈরি সাব কল করছি। প্রথমে আমার কল করেছি ReadInfo() সবাকে। এটা ব্যবহারকারীকে তথ্য ঢুকানোর জন্য ডায়লগ প্রদর্শন করবে। তথ্য ঢুকানো শেষ হলেই দ্বিতীয় সাব FillDGV(Info, DataGridView1) কল করা হবে। এ সাবের কাজ হলো তথ্যাদি গ্রিডে প্রদর্শন করা। ব্যস! স্ট্রাকচার ব্যবহারের উদাহরণ প্রোগ্রাম শেষ হলো। এবার চালু করে দেখার পালা। নিচের চিত্রটি দেখুন। লক্ষ করুন, NAME-এ ক্লিক করলে সমগ্র তথ্য আক্ষরিকভাবে সোর্ট হবে।



চিত্র -২৫: স্ট্রাকচার প্রোগ্রাম চলাকালীন চিত্র

পরের পরিচ্ছেদে আমরা ভিজুয়্যাল বেসিক.নেট এর আরো একটি গুরুত্বপূর্ণ বিষয়ের উপর উদাহরণসহ আলোচনা করবো। এ বিষয়টি হলো অন্যত্র কোড লিখে সংরক্ষণ ব্যবস্থা। এ পর্যন্ত আমরা যা করেছি তা সবই ফর্মের ক্লাস সাবরুটিনের ভেতর ছিলো। এখন একটি কোড ব্লকের উপর আলোচনা করবো যা লিখা হবে বাইরে- আলাদাভাবে। তবে অবশ্যই এটা যে কোনো জায়গায় ব্যবহারযোগ্য হবে। এই ব্যবস্থার নাম মডিউল [Module]।

৭ম পরিচ্ছেদ: *Separate code block in Program*

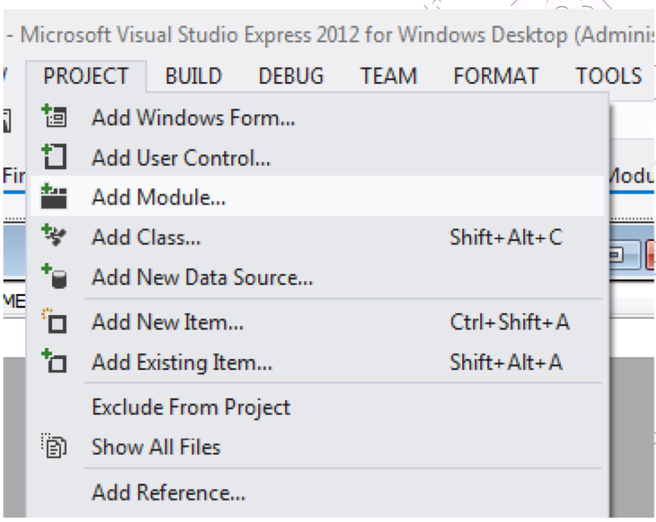
মডিউল ও ক্লাস : *Module & Class*

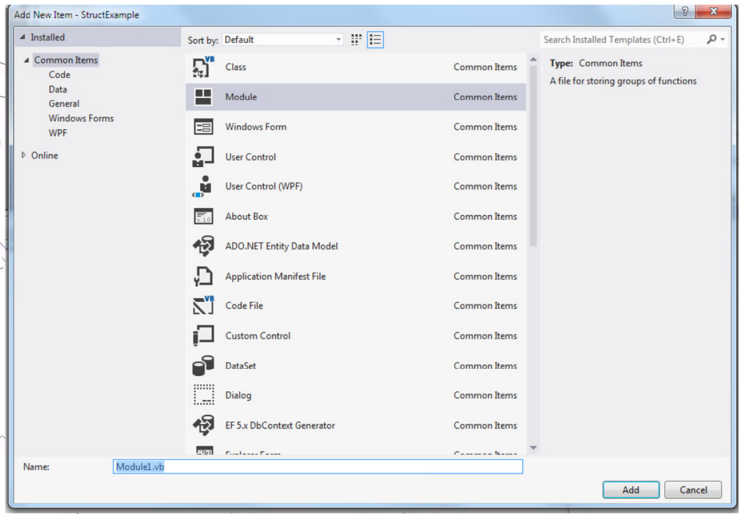
ভিজুয়্যাল বেসিক.নেটে আপনি আলাদাভাবে কোড ফাইল লিখে যে কোনো স্থানে সংরক্ষণ করতে পারেন। এটা খুব সুবিধাজনকও বটে। আপনি একই কোড বিভিন্ন প্রোগ্রামে ব্যবহার করতে চাইলে এ সুবিধা থাকাটা খুব জরুরীও।

মডিউল [Module]

মডিউলে শুধুমাত্র কোড থাকে। অর্থাৎ ফাংশন, সাব, স্ট্রাকচার ইত্যাদি ছাড়া আর কিছুই থাকবে না। এতে না আছে কোনো উইন্ডো, ফর্ম, বোতাম কিংবা গ্রাফিক্যাল ইউজার ইন্টারফেইস [GUI]। মডিউলে থাকবে শুধুমাত্র কোড।

প্রজেক্টে মডিউল যুক্ত করতে হলে প্রজেক্ট [Project] মেন্যুতে যেয়ে এড মডিউল [Add Module] চয়েজ করতে হবে। দেখুন নিচের ছবিটি (বায়ে)। Add Module চয়েজ করলে আসবে আরেকটি জানালা। ডানের চিত্রটি দেখুন।





এভাবে নতুন মডিউল বানিয়ে প্রজেক্টে যুক্ত করার পর মডিউলের কোড জানালা আসবে। আপনি এতে সাব, ফাংশন, স্ট্রাকচার, ইত্যাদি ডিক্লায়ার করতে পারেন। এসব বস্তু প্রোগ্রামের যে কোনো স্থান থেকে কল করে ব্যবহার করতে পারেন। সর্বোপরি এই মডিউল অন্য কোনো প্রজেক্টেও নিতে পারেন। আর মিডউল শব্দের অর্থই এটা। নিচে মডিউলের উদাহরণ কোড দেখিয়েছি। এর সাব ও ফাংশন কিংবা যে কোনো একটি প্রোগ্রামের অন্যত্র থেকে কল করতে পারেন। একটু স্টাডি করে নিন।

```
Public Module ThisModule
    Sub Main()
        Dim userName As String = InputBox("What is your name?") 'ask for user name
        MsgBox("User name is: " & userName) 'show it in message box
    End Sub
    Function Sq(SupNum As Double) As Double 'returns square of number
        Return SupNum * SupNum
    End Function
    ' Insert variable, property, procedure, and event declarations.
End Module
```

মডিউল ও ক্লাসের ব্যবহার এ পরিচ্ছেদের শেষে একটি উদাহরণে দেখিয়েছি।

৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : *Module & Class*

ভিজুয়্যাল বেসিক.নেট -এ মডিউলের বৈশিষ্ট্যাবলী:
১. এগুলো ভিজুয়্যাল বেসিক ফাইল।
২. এগুলোর মধ্যে প্রোগ্রামের সর্বত্র ব্যবহারযোগ্য সাব, ফাংশন ইত্যাদি সংরক্ষণ করা যায়।
৩. এগুলো ব্যবহার করলে বড় প্রোগ্রাম নিয়ন্ত্রণ ও রক্ষণাবেক্ষণ অনেকটা সহজ হয়।
৪. প্রোগ্রাম বুঝা সহজ হয়।
৫. এগুলো প্রোগ্রামে থাকলেও ব্যবহার করা ও না করার এখতিয়ার আছে।
৬. একাধিক মডিউল যে কোনো প্রোগ্রামে ব্যবহার করা যেতে পারে।
৭. মডিউল অপর কোনো প্রোগ্রামে নেওয়া যেতে পারে।

আমরা এবার আরো একটি একক কোড উইনিট নিয়ে আলোচনা করবো। এটা অনেকের মতে মডিউল থেকেও উন্নতমানের। এই বস্তুর নাম ক্লাস [Class]।

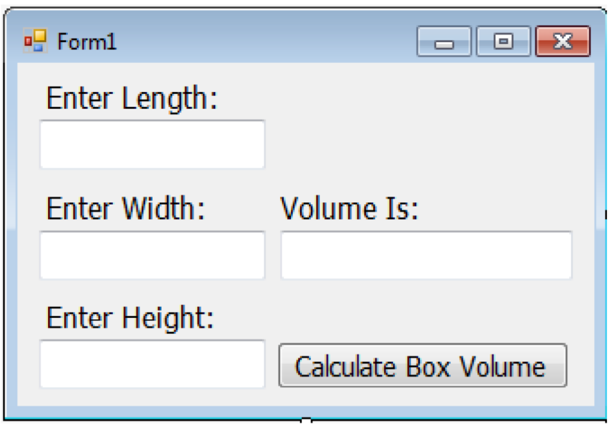
ক্লাস [Class]

ক্লাস অনেকটা স্ট্রাকচারের মতো কাজ করে। অতিরিক্ত সুযোগ হলো এতে সাবরুটিন ও ফাংশন যুক্ত করা যায়। ক্লাসের কোড পুনর্ব্যবহার করাও সম্ভব। এই ক্লাসের ভেতরই উইন্ডোজ ফর্ম প্রোগ্রামের কোড লিখিত থাকে। যেমন ফর্মের ক্লাস ডিক্লারেশন থাকে নিম্নরূপ:

```
Public Class Form1
    'Here all the subs and functions
End Class
```

অনুরূপ আমরা নিজেই আরো ক্লাস ডিক্লারেশন করতে পারি। নতুন ক্লাস বানানোর পদ্ধতি নিম্নরূপ:
১. প্রথমে প্রজেক্ট [Project] মেন্যুতে যেয়ে এড ক্লাস [Add Class] চয়েজ করুন।
২. ক্লাসের নাম দিন।
৩. সংরক্ষণ করুন। ক্লাস মডিউলের মতো আলাদা ফাইলে সংরক্ষিত হবে।

নিজের তৈরি ক্লাসের ব্যবহার সঠিকভাবে বুঝার জন্য এবার আমরা একটি প্রজেক্ট তৈরি করবো। আমাদের এ প্রোগ্রামের উদ্দেশ্য হলো একটি বাক্সের ভলিউম বের করে প্রদর্শন করা। এই প্রোগ্রাম যদিও অন্যভাবে সহজে লেখা যাবে তথাপি আমরা এটা একটি মডিউলের ভেতর ক্লাস ডিক্লারেশন দ্বারা করতে চাই। নিচের পদক্ষেপগুলো একে একে অনুসরণ করে প্রজেক্ট সৃষ্টি করুন ও কোড লিখুন।
১. নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন ModClassExample।
২. ফর্মের মধ্যে যুক্ত করুন ৪টি টেক্সট্ বাক্স ও একটি বোতাম। প্রত্যেক টেক্সট্ বাক্সের উপরে যুক্ত করুন একেকটি লেবেল। ডানের চিত্রের মতো পুরো ফর্মটি ডিজাইন করুন। ফর্মের FormBorderStyle হবে Fixed3D এবং MaximizeBox হবে False।
৩. প্রত্যেক লেবেল ও টেক্সট্ বাক্সের ফন্ট সাইজ হবে ১২।
৪. ডানের টেক্সট্ বাক্সের প্রপার্টিতে যেয়ে চয়েজ করুন: ReadOnly [True]। ব্যস! এ বাক্সে আর কিছু লেখা যাবে না। কারণ আমরা এতে শুধুমাত্র ফলাফল প্রদর্শন করবো।
৫. TextBox1, TextBox2 ও TextBox3 এর KeyPress Event সাবে [পরের পৃষ্ঠার] এই কোড লাইনগুলো যথাক্রমে লিখে দিন। এর ফলে এসব বাক্সে সংখ্যা ছাড়া আর কিছুই লিখা সম্ভব হবে না।



৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : *Module & Class*

```
'this three subs makes sure only decimal numbers are entered into textBox1, TextBox2 & TextBox3
Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
 Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox2.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
 Private Sub TextBox3_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox3.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
```

এবার প্রোগ্রামটি রান করে দেখুন। প্রথম তিনটি বাক্সে সংখ্যা ছাড়া অন্য কিছু লিখার চেষ্টা করুন। পারবেন না। তবে সংখ্যা [দশমিক বিন্দুসহ] ঠিকই লিখতে পারবেন। উপরের কোডগুলো গুরুত্বপূর্ণ। সংখ্যা ছাড়া অন্য কিছু থাকলে প্রোগ্রাম ক্রাশ করবে- যা আমাদের ইচ্ছে নেই।

৬. প্রোগ্রাম বন্ধ করে প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Class। ক্লাসের নামকরণ করুন: ClassBox.vb।

৭. ক্লাসের কোড হবে নিম্নরূপ:

```
Public Class ClassBox
    Public length As Double   ' Length of a box
    Public width As Double   ' Width of a box
    Public height As Double   ' Height of a box
End Class
```

৭. এবার প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Module। মডিউলের নামকরণ করুন: BoxMod.vb। কোড হবে:

```
Module BoxMod
   'define Volume Function here / public function available anywhere in the programme
   Public Function Vol(SupVL As Double, SupVW As Double, SupVH As Double) As Double
      Return SupVL * SupVW * SupVH
   End Function
End Module
```

৮. ডবল ক্লিক করুন Calculate Box Volume বোতামে। প্রদর্শিত সাবে লিখুন:

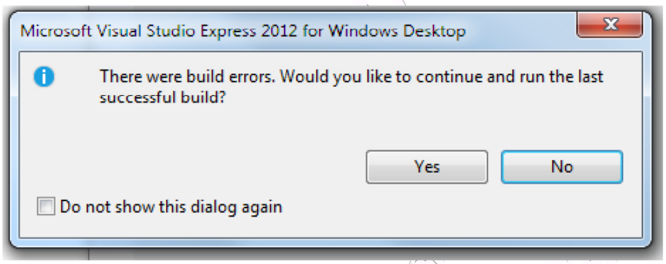
```
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      'make sure user entered numbers
      If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Then
         Exit Sub
      End If
      Dim Cl As New ClassBox 'declare a class type
      Cl.length = TextBox1.Text 'get value
      Cl.width = TextBox2.Text 'get value
      Cl.height = TextBox3.Text 'get value
     'Make sure they are numbers - in case user enters just space. Although in our example space not possible
     If IsNumeric(Cl.length) = False Or IsNumeric(Cl.width) = False Or IsNumeric(Cl.height) = False Then
         Exit Sub
      End If
      Dim ResVal As Double = Vol(Cl.length, Cl.width, Cl.height)
      TextBox4.Text = ResVal
   End Sub
```

৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : ***Module & Class***

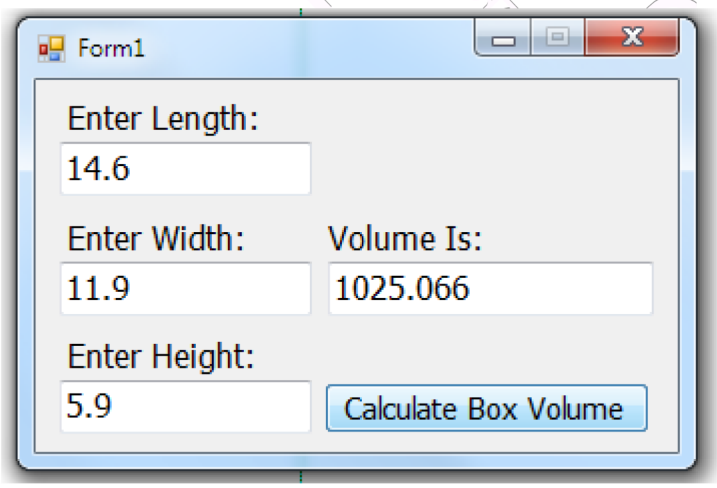
Compile and Test

ইতোমধ্যে আমরা প্রজেক্ট Start করে বার বার প্রোগ্রাম চালিয়ে পরীক্ষা করেছি। বাস্তবে Start মেন্যুটি চয়েজ করার পর কী হয়? সকল কোড প্রথমে এক্সপ্রেস ২০১২ বাইনারি কোডে পরিণত করে- একে বলে Compile। বাইনারি কোড মানে ০ এবং ১। আর ০ মানে সুইচ অফ এবং ১ মানে সুইচ অন। কম্পাইলার প্রথমে কোডের প্রতিটি লাইন একে একে পরীক্ষা করে দেখে এতে কোনো ভুল বা সিনটেক্সট এ্যারোর আছে কি না। যদি থাকে তাহলে কম্পাইলের পূর্বে একটি ওয়ার্নিং ডায়লগ আসবে। ডায়লগটি নিম্নরূপ:



সাধারণত আপনি 'No' বোতামে ক্লিক করবেন। তখন ভুলটি কোথায় হয়েছে তা-ও দেখাবে। ঠিক করার পর আবার Start Menu চয়েজ করে কম্পাইল করবেন। আপনার প্রোগ্রাম সম্পূর্ণ হয়ে গেলে কম্পাইল করে একটি **.exe** ফাইল তৈরি করবেন। এই কম্পাইল করা ফাইলই হলো এ্যাপলিকেশনের সর্বশেষ রূপ। একে পরিবর্তন করার কোনো সুযোগ আর থাকবে না। অন্যকথায় ডিস্ট্রিবিউশন পরে ব্যবহারকারী এই ফাইলে ডবল ক্লিক করে প্রোগ্রাম চালাবে। ব্যস! ভুল ত্রুটি সংশোধন একমাত্র আপনার কাজ। একে বলে ডিবাগিং। আপনি পরবর্তী ভার্শনে সংশোধিত আরেকটি **.exe** ফাইল [প্রয়োজনে Installer-সহ] পুনঃরিলিজ করবেন।

আপনার প্রোগ্রাম যদি বেশি জটিল হয় তাহলে বিল্ড মেন্যু [BUILD] চয়েজ করে আপনি 'Publish' [আপনার প্রোগ্রামের নাম]' চয়েজ করবেন। এরপর প্রদর্শিত নির্দেশনাবলী মুতাবিক পাবলিশ করে সংরক্ষণ করবেন। যাক, এ ব্যাপারে গ্রন্থের শেষের দিকে আরো বিস্তারিত আলোচনা হবে। এখন আমরা এইমাত্র সম্পন্ন ModClassExample প্রোগ্রামটি Start করে টেস্ট করবো। দেখুন নিচের চিত্রটি।



পরের পরিচ্ছেদে আমরা Sub এবং Function নিয়ে উদাহরণসহ অতিরিক্ত কিছু আলোচনার প্রয়াস পাচ্ছি। এর কারণ হলো, সকল ভিজুয়্যাল বেসিক.নেট মূলত এ দু'টোকে নিয়েই তৈরি হয়। সুতরাং এগুলো সম্পর্কে যা-ই আমরা জানবো তা-ই লাভজনক হবে।

৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

আমরা ইতোমধ্যে যাকিছু কোডিং করেছি তা সবই ছিলো মূলত সাব ও ফাংশন। হ্যাঁ, আমরা মডিউল এবং ক্লাস সম্পর্কেও জেনেছি। কিন্তু এগুলো আলাদা কোড ব্লক ছাড়া আর কিছু নয়। এগুলোর ভেতরও থাকে সাব এবং ফাংশন। সুতরাং সাব ও ফাংশন সম্পর্কে বেশি জানার অর্থ হলো .নেট কোডিং সম্পর্কে আরো বেশি অবগত হওয়া।

Writing Sub Procedure -সাব প্রসিডিউর লেখা

টেকনিক্যালী সাবরুটিনকে সাব প্রসিডিউর বলে। প্রত্যেক সাব প্রসিডিউর কল করা হয় তার নাম দিয়ে। এ কল হতে পারে অন্য কোনো ইভেন্ট সাব প্রসিডিউর থেকে কিংবা অন্য কোনো ইউজার-ডিফাইন্ড [অর্থাৎ আপনার বানানো] সাব প্রসিডিউর থেকে। এরা এক বা একাধিক তথ্য অথবা আর্গুমেন্ট গ্রহণ করে এর ওপর কাজ করতে পারে। কাজ থেকে ফলাফল ফেরৎ পাঠাতেও পারে। কিন্তু ফাংশনের মতো নামের সাথে ভ্যালু বা মান রিটার্ন করতে পারে না। অন্যকথায় সাবরুটিন ব্যবহৃত হয়: ১. ইনপুট (প্রদত্ত ভ্যালু) গ্রহণ বা প্রসেস করতে, ২. আউটপুট (ফলাফল) প্রদর্শন করতে ও ৩. প্রোপার্টিজ [বৈশিষ্ট্যাবলী] সেট করতে।

এবার একটি উদাহরণ দেখবো। সাব প্রসিডিউর প্রায়ই ব্যবহৃত হয় ইনপুট নিয়ন্ত্রণে। এই উদাহরণে আমরা 'AddName' নামক একটি Sub Procedure তৈরি করবো। এটা ব্যবহারকারীকে লেখা প্রবেশের জন্য অনুরোধ জানাবে এবং তা টেক্সট বাক্সে প্রদর্শন করে দেখাবে। প্রসিডিউরটি একটি মডিউলে থাকবে। ফলে একে একাধিকবার কল করা যাবে। মূলত পুরো প্রেগ্রামের যে কোনো স্থান থেকে একে কল করে ব্যবহার করা যেতে পারে।

১. একটি নতুন প্রজেক্ট তৈরি করুন। এর নামকরণ করুন: 'My Text Box Sub'।
২. ফর্মে দু'টি টেক্সট্ বক্স যুক্ত করুন।
৩. বাক্স দু'টোর উপরে দু'টি লেবেল যুক্ত করুন। এগুলোতে থাকবে ডিপার্টমেন্টের নাম।
৪. মোট তিনটি বোতাম যুক্ত করুন। প্রথম দু'টি যথাক্রমে টেক্সট্ বাক্স ১ ও টেক্সট্ বাক্স ২ এর নিচে রাখুন। বাকিটি রাখুর ফর্মের নিচের দিকে।
৫. নিচের টেবিল মুতাবিক বিভিন্ন বস্তুর প্রোপার্টি সেট করুন:

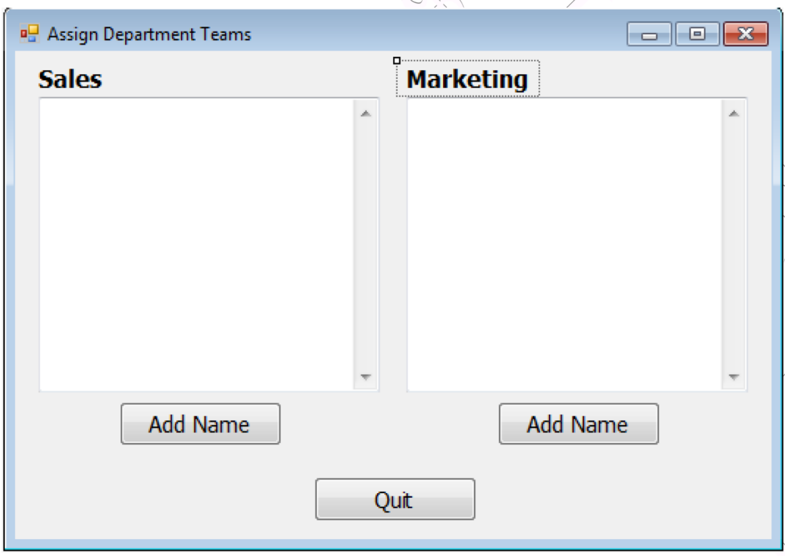
বস্তু	প্রোপার্টি	সেটিং
TextBox1	Multiline	True
	Name	txtSale
	ReadOnly	True
	ScrollBars	Vertical
	TabStop	False
	Font Size	12
TextBox2	All Same accept Name txtMkt	
Label1	Font	Bold
	Font Size	12
	Name	lblSales
	Text	"Sales"
Label1	Font	Bold
	Size Size	12
	Name	lblMkt
	Text	"Marketing"

৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

বস্তু	প্রোপার্টি	সেটিং
Button1	Name	btnSales
	Text	"Add Name"
	Font Size	11
Button2	Name	btnMkt
	Text	"Add Name"
	Font Size	11
Button3	Name	btnQuit
	Text	"Quit"
	Font Size	11
Form1	Text	"Assign Department Teams"

সব ঠিকমতো হলে ফর্মটির ডিজাইন হবে নিচের চিত্রের মতো।



এখন আমরা প্রজেক্টে একটি মডিউল যুক্ত করবো। আর এতেই লিখবো আমাদের সর্বত্র ব্যবহারযোগ্য AddName সাবরুটিনটি।

৬. প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Module। নতুন মডিউল তৈরি হবে। এটার নামকরণ যা আছে তা-ই রেখে দিন।

৭. মডিউল তৈরির পরই এর কোড জানালা খুলবে। এই কোড জানালায় থাকবে নিচের কোড ব্লকটি:

```
Module Module1
    'Our Sub here
End Module
```

আমরা যে সাব প্রসিডিউর লিখবো তা এই ব্লকের ভেতরই থাকবে।

৭. নিচের কোডটি হবে মডিউলের মধ্যে।

```
Module Module1
    Sub AddName(ByVal Team As String, ByRef ReturnString As String)
        Dim Prompt, Nm, WrapCharacter As String
        Prompt = "Enter a " & Team & " employee."
        Nm = InputBox(Prompt, "Input Box")
        WrapCharacter = Chr(13) + Chr(10)
        ReturnString = Nm & WrapCharacter
    End Sub
End Module
```

ব্যাখ্যা: এই প্রসিডিউর কল হলে দু'টি আর্গুমেন্ট গ্রহণ করবে। এগুলো হলো Team, স্ট্রিং টাইপ ডিপার্টমেন্টের নাম এবং ReturnString, একটি শূন্য স্ট্রিং ভেরিয়েবল যাতে থাকবে কর্মকতার নাম। লক্ষ্য করুন, ReturnString-কে

৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

ByRef কিওয়ার্ডসহ ডিক্লায়ার করা হয়েছে। এর ফলে প্রসিডিউরে এই আর্গুমেন্টে কোনো পরিবর্তন হলে তা কলিং রুটিনে ফিরে দেওয়া হবে। প্রথম লাইনে আমরা তিনটি স্ট্রিং ভেরিয়েবল ডিক্লায়ার করেছি। এগুলো ২য়, ৩য় ও ৪র্থ লাইনে ইনিশিয়েলাইজ করা হয়েছে। শেষের লাইনে রিটার্নস্ট্রিংকে ফর্মেট করেছি। ব্যস! এটুকুই। এই প্রসিডিউর ব্যবহারকালে বুঝতে পারবেন এটা কিভাবে কাজ করে।

৮. পুনরায় ফর্ম ডিজাইন প্রদর্শন করুন। ডবল ক্লিক করুন ফর্মের Add Name বোতামে। বোতামের ক্লিক ইভেন্ট প্রসিডিউর খুলবে। এতে যুক্ত করুন নিচের কোডটি।

```
'calls AddName Procedure defined in Module1
Private Sub btnSales_Click(sender As Object, e As EventArgs) Handles btnSales.Click
    Dim SalesPosition As String = ""
    AddName("Sales", SalesPosition)
    txtSale.Text = txtSale.Text & SalesPosition
End Sub
```

ব্যাখ্যা: AddName প্রসিডিউর কলে দু'টি আর্গুমেন্ট আছে: Sales এবং SalesPosition। শেষের লাইনের কাজ হলো রেফারেন্সের মাধ্যমে প্রেরিত আর্গুমেন্ট ব্যবহার করে txtSale টেক্সট বাক্সে লেখা যুক্ত করা। কনক্যারেটশন অপারেটর & এর কারণে নতুন নাম টেক্সটের শেষে যুক্ত হবে। যাক, প্রোগ্রাম চলাকালে বিষয়টি আরো পরিষ্কার হবে।

৯. এবার ডবল ক্লিক করুন ডানের Add Name বোতামে। কোড এডিটরে বোতামের ক্লিক ইভেন্ট সাব দেখাবে। এই সাবের ভেতর লিখুন:

```
Private Sub btnMkt_Click(sender As Object, e As EventArgs) Handles btnMkt.Click
    Dim MktPosition As String = ""
    AddName("Marketing", MktPosition)
    txtMkt.Text = txtMkt.Text & MktPosition
End Sub
```

ব্যাখ্যা: এই বোতামের কোড প্রায় একই। পার্থক্য হলো মডিউলের AddName প্রসিডিউরে "Marketing" পাটিয়ে দেয় এবং আপডেট করে txtMkt টেক্সট বাক্স।

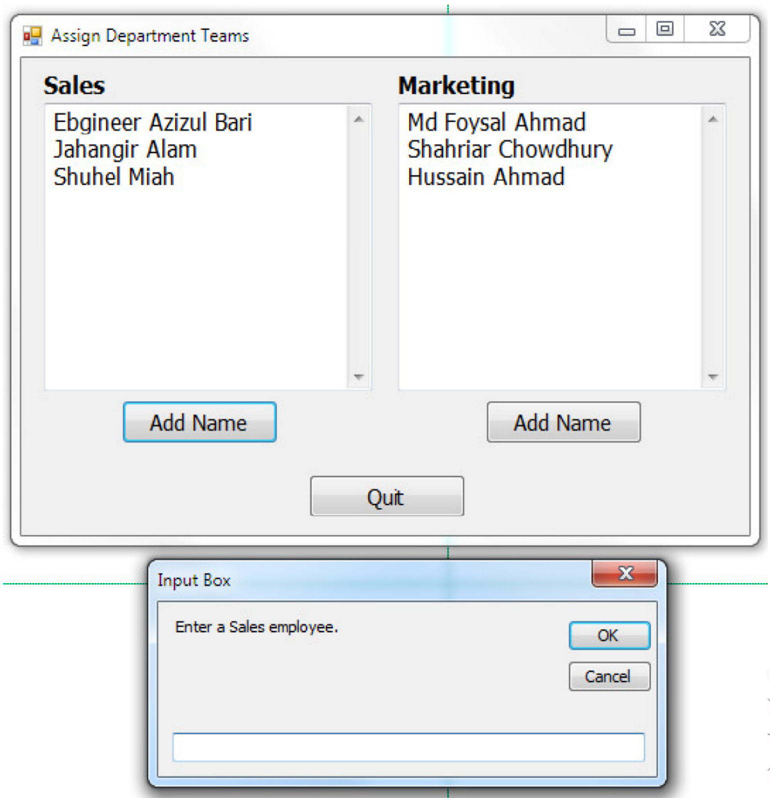
১০. এবার ডবল ক্লিক করুন Quit বোতামে। এর ক্লিক ইভেন্টে লিখুন:

```
'This sub quits the application
Private Sub btnQuit_Click(sender As Object, e As EventArgs) Handles btnQuit.Click
    End
End Sub
```

১১. এবার Start মেন্যুতে ক্লিক করে প্রোগ্রাম পরিচালনা করুন। উভয় Add Name বোতামে ক্লিক করে দেখুন। পরের পৃষ্ঠায় প্রোগ্রাম চলাকালীন চিত্র দেখিয়েছি। এ উদাহরণ দ্বারা এটাই দেখানো হলো যে, একটি মাত্র সাব প্রসিডিউর ব্যবহার করে দুই বা ততোধিক ইনপুট কাজ করা যায়। এই মৌলিক টেকনিক কাজে লাগিয়ে বার বার ব্যবহারযোগ্য শক্তিশালী সাব এবং (পরবর্তীতে আলোচ্য) ফাংশন সৃষ্টি করতে পারেন আপনার প্রোগ্রামের জন্য।

৮ম পরিচ্ছেদ: ***More on Subs & Functions***

সাব ও ফাংশন : ***Subs & Functions***



প্রত্যেকবার Add Name বোতামে টিপ দিলেই নিচের ইনপুট বক্স ডায়লগ আসবে এবং যে নাম লিখবেন তা টেক্সট বাক্সে প্রদর্শিতও হবে। সুতরাং আপনি কোম্পানির সেল্‌স ও মার্কেটিং কর্মকর্তাদের একটি নামের লিস্ট বানাতে সক্ষম হলেন। হয়তো ভাবছেন এটা সংরক্ষণ করা যায় কিভাবে? এ ব্যাপারে ফাইল ম্যানেজমেন্টের উপর আলোচনাকালে ব্যাখ্যা আসবে। আপাতত এটাই জানা উদ্দেশ্য যে, একই প্রসিডিউর বার বার ব্যবহার করা সম্ভব।

Writing Function Procedure - ফাংশন প্রসিডিউর লেখা

আমরা জানি ফাংশন প্রসিডিউর হলো একগুচ্ছ ভিজুয়্যাল বেসিক.নেট কোড যা লেখা হয় Function এবং End Function এর মাঝখানে। এই কোডগুলোর কাজ হলো কোনো অর্থপূর্ণ কর্ম সম্পাদন। যেমন: টেক্সট প্রসেস করা, ইনপুট নিয়ন্ত্রণ কিংবা কোনো মান বেরকরণ ইত্যাদি কাজে লাগাতে ফাংশন কল করতে হয়। ফাংশনে প্রয়োজনে আর্গুমেন্ট বা প্যারামিটার যুক্ত করা যায়। কলের সময় ব্র্যাকেটের ভেতর এসব প্যারামিটারের নাম দিতে হবে। মূলত কিম্পউটারে অসংখ্যা রেডি-মেইড ফাংশন প্রসিডিউর আছে। প্রত্যেক এ্যাপলিকেশনই প্রয়োজনমাফিক এসব ফাংশন ব্যবহার করে থাকে। আমরাও ব্যবহার করেছি। যেমন: MsgBox, InputBox ইত্যাদি। নিজের তৈরি ফাংশন ও এসব সিস্টেম ফাংশনে মূলত তেমন কোনো পার্থক্য নেই। ফাংশনের আরেকটি বৈশিষ্ট্য হলো এগুলো যদি আমরা আলাদা মডিউলে ডিফাইন করি তাহলে সমগ্র প্রোগ্রামব্যাপী তা ব্যবহার করতে পারি। একই কথা সাব প্রসিডিরের ক্ষেত্রেও সত্য।

ফাংশনের আরেকটি উদাহরণ উপস্থাপনের পূর্বে তিনটি ব্যাপার পরিষ্কার করা প্রয়োজন বোধ করছি। যে কোনো ফাংশনের সিনটেক্সটে যা থাকা জরুরী তার মধ্যে এ তিনটি গুরুত্বপূর্ণ: ১. ফাংশনের নাম, ২. রিটার্ন টাইপ (যদি না থাকে তাহলে ভিজুয়্যাল বেসিক ‘অবজেক্ট’ টাইপ রিটার্ন করবে] ও ৩. ফাংশন স্টেটম্যান্টস্। যেমন:

নিচের ফাংশন দু’টি টেক্সট মান হিসাব করে যোগফল রিটার্ন করে। রিটার্ন মান হবে সিঙ্গুল ডাটা টাইপ।

```
Function TotalTax(ByVal Cost as Single) As Single
    Dim StateTax, CityTax As Single
    StateTax = Cost * 0.05 'State tax is 5% 'value of cost comes from calling line
    CityTax = Cost * 0.015 'City tax is 1.5%
    TotalTax = StateTax + CityTax 'to return value function name is used.
    [Return StateTax + CityTax] 'রিটার্ন কিওয়ার্ড দ্বারাও মান ফেরৎ পাঠানো যায়।
End Function
```

কলিং: উক্ত ফাংশন কল করার উদাহরণ: TextBox1.Text = TotalText(400)।

লক্ষ করুন: ফাংশন নাম দ্বারা কিংবা **Return** কিওয়ার্ড দ্বারা মান রিটার্ন করা যায়। তবে উভয়টা ব্যবহার করা যাবে না।

যাক, এখন আমরা একটি নতুন প্রজেক্ট করবো এবং ফাংশনের ব্যবহার দেখাবো।

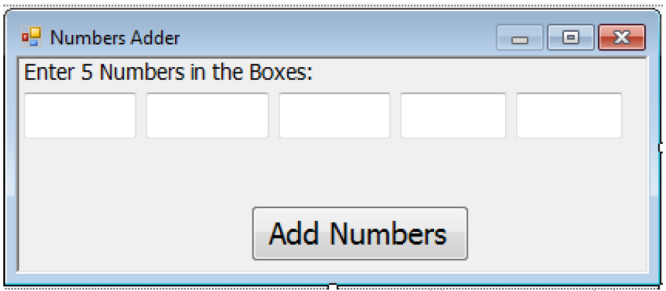
৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

NumbersAdder উদাহরণ

ফাংশনের ব্যবহার দেখাতে যেয়ে আমরা একটি নতুন প্রজেক্ট তৈরি করবো। আমাদের উদ্দেশ্য হলো ব্যবহারকারী ৫টি সংখ্যা পাঁচটি টেক্সট্ বাক্সে লিখবেন। এগুলোর যোগফল একটি বোতামে ক্লিক করে বের করা।

১. একটি নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন: NumbersAdder।
২. ফর্মে থাকবে দু'টি লেবেল, ৫টি টেক্সট্ বাক্স ও ১টি বোতাম।
৩. ফর্ম ডিজাইন করুন নিচের ছবির মতো।



৪. ফর্মের Text প্রোপার্টি হবে Numbers Adder। FormBorderStyle প্রোপার্টি হবে Fixed3D এবং MaximizeBox প্রোপার্টি হবে False।
৫. label1 এর Text প্রোপার্টি হবে Enter 5 Numbers in the Boxes:।
৫. Button1 এর Text প্রোপার্টি হবে Add Numbers।
৬. উভয় লেবেলের ফন্ট সাইজ হবে ১০ এবং সকল টেক্সট্বাক্স ও বোতামের ফন্ট সাইজ হবে ১৪।

কোডিং

প্রথমত আমাদেরকে প্রতিটি টেক্সট্ বাক্সে যাতে দশমিক কিংবা আস্ত সংখ্যা ছাড়া আর কিছু লেখা যায় না তা নিশ্চিত করতে হবে। মনে আছে, ভেলিডেশন কোড? নিচের কোড লাইনটি প্রত্যেক টেক্সট্ বাক্সের KeyPress ইভেন্ট হ্যান্ডলারে লিখতে হবে:

```
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
```

এবার নিচের ফাংশনটি ফর্মের ক্লাসের ভেতর লিখতে হবে:

```
'this function takes a list of numbers separated by commas and returns total of the numbers
Public Function calcSum(ByVal ParamArray args() As Double) As Double
    calcSum = 0
    If args.Length <= 0 Then Exit Function
    For i As Integer = 0 To UBound(args, 1)
        calcSum += args(i)
    Next i
End Function
```

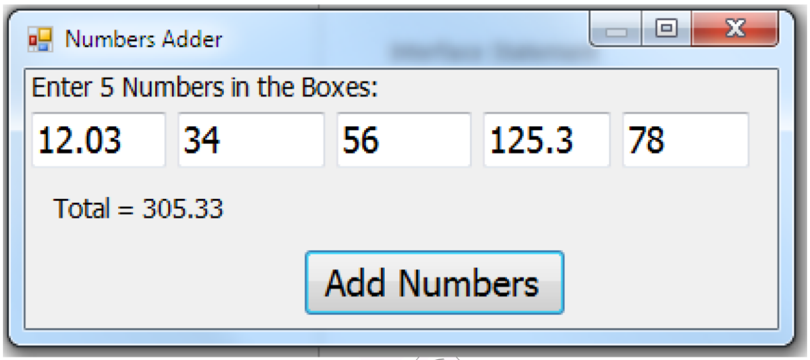
লক্ষ করুন: ByVal ParamArray args() As Double দ্বারা আমরা বুঝাচ্ছি যে, ফাংশনের প্যারামিটার বা আর্গুমেন্ট হবে একটি এ্যারে ভেরিয়েবল। এর টাইপ ও রিটার্ন টাইপ হবে ডাবল। এই ফাংশনের মান ফাংশনের নাম দ্বারাই হবে। এজন্য আমরা প্রথম লাইনে নাম লিখে তার মান ০ বানিয়েছি। দ্বিতীয় লাইনে আমরা নিশ্চিত করছি এ্যারের মধ্যে শূন্য কিংবা এর কম পরিমাণ ইলেমেন্ট থাকলে কিছু না করতে। এরপর ফর-নেস্ট লুপ দ্বারা এ্যারের প্রতিটি ইলেমেন্ট তথা ৫টি সংখ্যার যোগফল বের করে রিটার্ন করছি কলিং লাইনে। আর কলিং হচ্ছে নিচের সাবে। ব্যাস!

```
 'this is the calling sub
 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    On Error GoTo ExitLabel 'error handling/ if user leaves on or more text field empty
    Dim a As Double = TextBox1.Text : Dim b As Double = TextBox2.Text 'put numbers into variables
    Dim c As Double = TextBox3.Text : Dim d As Double = TextBox4.Text 'put numbers into variables
    Dim f As Double = TextBox5.Text 'put numbers into variables
    Dim SumOfNumber = calcSum(a, b, c, d, f) 'function call
    Label2.Text = "" 'empty out the result
    Label2.Text = "Total = " & SumOfNumber 'show result
    ExitLabel: 'error jump
End Sub
```

৮ম পরিচ্ছেদ: ***More on Subs & Functions***

সাব ও ফাংশন : *Subs & Functions*

এবার প্রোগ্রামটি রান করে দেখা যাক, কেমন? নিচের চিত্রে প্রোগ্রাম চলাকালীন জানালা দেখা যাচ্ছে।



লক্ষ করুন: টেক্সট বাক্সগুলোতে আস্ত এবং ডেসিমেল সংখ্যা ঢুকাতে পারবেন। তবে একটিও বাক্সে কিছু না থাকলে ফলাফল কিছুই পাবেন না।

যাক, আমরা এই সলিউশন দ্বারা দেখালাম, ফাংশনে এ্যারে ভেরিয়েবল আর্গুমেন্ট হিসাবে ব্যবহার সম্ভব। আমাদের ফাংশনের আর্গুমেন্ট বা প্যারামিটার হিসাবে আমরা পাঁচটি সংখ্যা দেখিয়েছি: calcSum(a, b, c, d, f)। বাক্সের সংখ্যাকে এ্যারের আইটেম হিসাবে পরিবর্তন করে ফাংশন কল করে ফলাফল নিয়ে এসেছি। একটা ব্যাপার হয়তো আপনার নিকট পরিষ্কার নয়। তাহলো এই কোড লাইনটি: For i As Integer = 0 To UBound(args, 1)। বাস্তরে ব্যাপারটা তেমন জটিল নয়। কিওয়ার্ড এর অর্থ হলো Upper Bound। অর্থাৎ এ্যারের ব্যাপ্তি হবে ১ থেকে [এক্ষেত্রে] ৫। অন্যকথায় লুপের মধ্যে **i** যখন ৪ হবে তখন UBound(args, 1) এর ফলও হবে ৫। সুতরাং ৫ বার মাত্র ফর-নেক্সট লুপ ঘুরবে এবং সংখ্যাগুলো যোগ করে যাবে। এখানে এই টেকনিক ব্যবহারের মূল কারণ হলো এ্যারোকে প্রয়োজনে ছোট বড় করা যাবে। ফাংশনে কোনো কাজ করতে হবে না।

৯ম পরিচ্ছেদ: *Ready-made functions and methods*

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

এ পরিচ্ছেদে আমরা বানানো কিছু ফাংশন ও মেথড নিয়ে উদাহরণসহ আলোচনা করবো। ভিজুয়্যাল বেসিক.নেট এর সৃষ্টিকারীরা প্রোগ্রামারদের সুবিধার জন্য প্রচুর ফাংশন ও মেথড তৈরি করে সিস্টেমে যুক্ত করে দিয়েছেন। এগুলো বিভিন্ন নাম ও নিয়মে ব্যবহার করা যায়। প্রথমে আমরা এরূপ কিছু ফাংশন উদাহরণসহ নিম্নে উপস্থাপন করছি।

ক. String Functions - স্ট্রিং ফাংশন্স -লেখা সম্পর্কিত ফাংশন

১. Split - মনে করুন: 'বাগানে অনেক আকর্ষণীয় ফুল ফুটেছে।' এ বাক্যটির প্রত্যেক শব্দ আলাদা করবেন। কোডিংয়ের মাধ্যমে এরূপ করতে চাইলে আপনি ব্যবহার করতে পারেন Split ফাংশনটি।

উদাহরণ:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
 TextBox1.Text = "" 'clean text box
  Dim A_String As String = "evMv‡b A‡bK AvKl©Yxq dzj dz‡U‡Q|"
  'split the string based on spaces and put them in an Array variable
  Dim Words As String() = A_String.Split(New Char() {" "})
  'Display words in a text box in seperate lines
  For A = 0 To Words.Length - 1 'Words.Length gives number of words in Array
     TextBox1.Text = TextBox1.Text & Words(A) & vbCrLf
  Next
End Sub
```

লক্ষ করুন: Split ফাংশন দিয়ে লেখার মধ্যে মোট ক'টি শব্দ আছে তা সহজে বের করা যায়। যেমন:

```
Dim Words As String() = A_String.Split(New Char() {" "}
MsgBox(Words.Length) 'this will show number of words
```

২. Join -আলাদা স্ট্রিং ইচ্ছে মতো একত্রিত করা যায় এই ফাংশন দিয়ে। যেমন:

```
'Join function at work
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
   Dim An_Array(4) As String 'declare an array of 5 words
   An_Array(0) = "evMv‡b" : An_Array(1) = "A‡bK" : An_Array(2) = "AvKl©Yxq"
   An_Array(3) = "dzj" : An_Array(4) = "dz‡U‡Q|" 'put words into them
   Dim JoinText As String = String.Join(" ", An_Array) 'using space as delimiter
   TextBox1.Text = JoinText 'display result
End Sub
```

৩. Compare -দু'টি স্টিং এর মধ্যে পার্থক্য নির্ণয় সময় সময় জরুরী হয়ে ওঠে। ভিজুয়্যাল বেসিক এ ফাংশন তৈরি করেছে পার্থক্য নির্ণয়ের জন্য। যেমন:

```
Dim StA As String = "a" : Dim StB As String = "b"
Dim C As Integer = String.Compare(StA, StB) : MsgBox(C) 'gives -1
C = String.CompareOrdinal(StB, StA) : MsgBox(C) 'gives 1
C = StB.CompareTo(StA) : MsgBox(C) 'gives 1
C = "X".CompareTo("X") : MsgBox(C) 'gives 0
```

এখানে: -১ অর্থ প্রথমটি ছোট, ১ অর্থ দ্বিতীয়টি বড় এবং ০ অর্থ উভয়টি সমান। যে কোনো শব্দ বা বাক্যও সমান কি না এই ফাংশন দ্বারা পরীক্ষা করা যায়। পরের পৃষ্ঠায় আরেকটি উদাহরণ দেখুন।

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

```
Dim Word1 As String = "আসমান" : Dim Word2 As String = "প্রথম"
    'comapre
If String.Compare(Word1, Word2, False) = 0 Then
        MsgBox("String Match")
    Else
     MsgBox("String does not Match") 'this message appears
    End If
```

৪. Insert -এই ফাংশনের কাজ হলো স্ট্রিং এর ক্যারাক্টার পজিশনে প্রদত্ত স্ট্রিং ঢুকানো। যেমন:

```
' Insert the word [and].
    Dim names As String = "Romeo Juliet"
    Dim Shakespeare As String = names.Insert(6, "and ") ' 'and ' inserted in the space (character     'position
6: result: "Romeo and Juliet"
```

৫. Length -এই ফাংশনের কাজ হলো স্ট্রিং এর ভেতর ক'টি ক্যারাক্টার আছে তা নির্ণয় করা: যেমন:

```
    Dim value As String = "dotnet"
    Dim length As Integer = value.Length 'gives length = 6
    value += ".aziz"  'now value is 'dotnet.aziz'
    length = value.Length 'gives length = 11
```

৬. Remove -এটা দ্বারা কোনো স্ট্রিং এর প্রদত্ত পজিশন থেকে বাকি সবগুলো মুছে ফেলা যায়। যেমন:

```
    ' Completely incorrect input string.
      Dim value As String = "Engineer Azizul Bari wrote 'Visual Basic.Net Programming'. and.."
      'We need to remove ' and..' part, eg. everything after last SPACE
      Dim index As Integer = value.LastIndexOf(" "c)
      ' Remove everything starting at that position.
      value = value.Remove(index)
      TextBox1.Text = value
```

৭. Replace -এ ফাংশনের কাজ হলো যে কোনো স্ট্রিং - যেমন কোনো ডকুমেন্টের টেক্সটে একই শব্দ বা লাইন ইত্যাদি যতোবারই থাকুন না কেনো, তা এক সঙ্গে বদলানো। যেমন:

```
Dim FstString As String = "আমরা এই লেখার তাই শব্দ বদলাতে চাই। তাই, তাই"
      ঋৎঃঋঃৎরহম = ঋৎঃঋঃৎরহম.জবঢ়ষধপব("তাই", "")
      MsgBox(FstString) 'gives 'আমরা এই লেখার শব্দ বদলাতে চাই।'
```

৮. Substring -এটা একটি সচরাচর ব্যবহৃত ফাংশন। এর কাজ হলো যে কোনো লেখা থেকে প্রদত্ত অক্ষরসমূহ সরিয়ে নিয়ে প্রদর্শন করা। যেমন:

```
          'Use this string literal for the demonstration.
           Dim literal As String = "Our Our country is Bangladesh." -আমরা 'Our' শব্দটি বের করবো।
           Dim substring As String = literal.Substring(0, 4) 'first word
          MsgBox(substring) 'will show - Our country is Bangladesh.
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৯. Contains -এটা দ্বারা জানা যায় একটি স্টিং এর মধ্যে আরেকটি স্ট্রিং আছে কি না। ফলাফল True বা False। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।" : Dim newString As Boolean =
literal.Contains("প্রথম")
    MsgBox("প্রথম: " & newString) 'result True : Dim newStringt As Boolean = literal.Contains
("আসমান") : MsgBox("আসমান: " & newStringt) 'result False
```

১০. InStr -এই ফাংশন দিয়ে একটি স্ট্রিং এর ভেতর আরো একটি স্ট্রিং আছে কি না পরীক্ষা করা যায়। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।"
Dim isString As Integer = InStr(literal, "বের", CompareMethod.Text)
MsgBox(isString) 'returns 24 -found at character position 24
isString = InStr(literal, "আমার", CompareMethod.Binary)
MsgBox(isString) 'retuns 0 -string not found
```

১১. Mid -এটি দিয়ে স্ট্রিং এর ভেতর থেকে কাঙ্ক্ষিত স্ট্রিং বের করা যায়। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।"
Dim NString As String = Mid(literal, 6, Len(literal))
MsgBox(NString) 'result-  থেকে প্রথম শব্দটি বের করবো।
```

লক্ষ করুন: Length ফাংশন ও উপরে ব্যবহৃত Len ফাংশন এর কাজ একই। প্রথমটি ভিজুয়্যাল বেসিক-কে ভিজুয়্যাল বেসিক.নেট -এ উন্নীতকরণকালে সংযুক্ত করা হয়েছে। তবে উভয়টিই কার্যকরী।

১২. একটিমাত্র ক্যারাক্টার বারবার লেখা: নিচের কোডটি এ কাজ করে।

```
Dim txt As New String("স"c, 5)
MsgBox(txt) -ফলাফল -সসসসস
```

১৩. ToUpper & ToLower -এ ফাংশন দ্বারা দু'টো শুধুমাত্র ইংরেজি অক্ষরসমূহ বড় এবং ছোট করা যায়। যেমন:

```
Dim word As String = "ViSuAL bASiC"
MsgBox(word.ToUpper) 'gives VISUAL BASIC
MsgBox(word.ToLower) 'gives visual basic
```

নোট: বাংলা ইউনিকোড লেখা হলে উভয় ফাংশন কাজ করে না। লেখা অপরিবর্তিত থাকবে।

১৪. স্ট্রিং -এর সকল অক্ষর আলাদা করে বের করা: নিচের কোডটুকু এ কাজ করবে। লক্ষণীয়- স্পেইস বা শূন্যস্থানকেও কোড ক্যারাক্টার হিসাবে সনাক্ত করে।

```
Dim AworD As String = "I am a programmer."
For i As Integer = 0 To AworD.Length - 1 'Or 0 to len(AworD) -1
    MsgBox(AworD.Chars(i)) 'shows each character of the sentence
Next
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

১৫. IsDigit -সময় সময় আমাদের জানতে হয় ক্যারাক্টার সংখ্যা [০-৯] কি না। এ ফাংশন দ্বারা তা জানা যায়। যেমন:

```
Dim EntTx As String = TextBox1.Text
    If EntTx <> "" Then
        MsgBox(Char.IsDigit(EntTx.Chars(2)))
    'যদি তৃতীয় অক্ষর সংখ্যা হয়- ফলাফল True। অন্যথায় False হবে।
End If
```

অনুরূপ আরো দু'টি ফাংশন আছে: (ক) IsNumeric -এটা পুরো লেখাটি সংখ্যা (ডেসিমেলসহ) কি না দেখবে। (খ) IsNumer(Char, Character Position) -এটা ঠিক IsNumeric-এর মতো, তবে চিহ্নিত একটিমাত্র ক্যারাক্টার ইউনিকোড হলেও সনাক্ত করবে। যেমন বাংলা ইউনিকোড ক্যারাক্টার ৯ এ ফাংশন সংখ্যা হিসাবেই সনাক্ত করবে। দৃষ্টান্ত:

```
Dim AString As String = "345.03"
If IsNumeric(AString) = True Then MsgBox("Yes") 'result Yes because whole string is a double
If Char.IsNumber(AString, 3) = True Then
    MsgBox("Yes")
Else
    MsgBox("No") 'result No because . isn't a number.
End If
AString = "৩৯৫.০৩"□বাংলা ইউনিকোড সংখ্যা□
□□□□□□□□If IsNumeric(AString) = True Then
    MsgBox("Yes")
Else
    MsgBox("No") 'result No because IsNumeric doesn't recognize unicode
End If
If Char.IsNumber(AString, 1) = True Then
    MsgBox("Yes") 'result Yes because index 1 = 9 [Remember: index always 0 based]
Else
    MsgBox("No")
End If
```

১৬. Trim, LTrim, RTrim -এ তিনটি ফাংশন যথাক্রমে লেখার উভয় দিকের, বায়ের ও ডানের শূন্যস্থান [স্পেইস] মুছে ফেলে। যেমন:

```
Dim AString As String = " 345.03 is a number. "
    AString = (Trim(AString)) 'both sides space erased
    AString = (LTrim(AString)) 'left side space erased
    AString = (RTrim(AString)) 'right side space erased
    TextBox1.Text = AString 'Clean text: '345.03 is a number.'
```

খ. Conversion Functions - কনভারশন ফাংশন্স -রূপান্তর সম্পর্কিত ফাংশন

১. Asc -এই ফাংশন দিয়ে যে কোনো ক্যারাক্টারের সংখ্যা কোড জানা যায়। যেমন:

```
Dim codeInt As Integer
' The following line of code sets codeInt to 65.
codeInt = Asc("A")
' The following line of code sets codeInt to 97.
codeInt = Asc("a")
' The following line of code sets codeInt to 65.
codeInt = Asc("Apple")
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

২. AscW -এই ফাংশন দিয়ে যে কোনো ইউনিকোড ক্যারাক্টারের সংখ্যা কোড জানা যায়। কোড সংখ্যার ব্যাপ্তি হলো ০ থেকে ৬৫৫৩৫। উদাহরণ:

```
AscW("প") 'returns 2474
AscW("ক") 'returns 2453
```

৩. Chr -এ ফাংশন সংখ্যা কোড থেকে ক্যারাক্টার বের করে। যেমন:

```
Chr -retuns character
Dim associatedChar As Char
' Returns "A".
associatedChar = Chr(65)
' Returns "a".
associatedChar = Chr(97)
' Returns ">".
associatedChar = Chr(62)
' Returns "%".
associatedChar = Chr(37)
```

৪. ChrW -এ ফাংশন সংখ্যা কোড থেকে ইউনিকোড ক্যারাক্টার বের করে। যেমন:

```
ChrW(2474) returns 'প'
ChrW(2453) returns 'ক'
```

৫. Format -এটি সাধারণত তারিখ ফরম্যাট করার জন্য ব্যবহৃত হয়। যেমন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim TestDateTime As Date = #1/27/2001 5:04:23 PM#
    Dim TestStr As String
    TestStr = Format(Now(), "Long Time") ' Returns current system time in the system-defined long time format.
    TextBox1.Text += TestStr & vbCrLf ' Also returns current system date in the system-defined long date format, using the single letter code for the format.
    TestStr = Format(Now(), "D")
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "h:m:s") ' Returns the value of TestDateTime in user-defined date/time formats. Returns "5:4:23".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "hh:mm:ss tt") ' Returns "05:04:23 PM".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "dddd, MMM d yyyy") ' Returns "Saturday, Jan 27 2001".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "HH:mm:ss") ' Returns "17:04:23".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(23) ' Returns "23"
    TextBox1.Text += TestStr
End Sub
```

বায়ের প্রোগ্রাম রান করার ফলাফল:

```
12:50:54 PM
Wednesday, August 10, 2016
5:4:23
05:04:23 PM
Saturday, Jan 27 2001
17:04:23
23
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

অবশ্য ফাংশন দ্বারা ব্যবহারকারীর ইচ্ছেমাফিক কাজও করা যায়। যেমন নিচের কোড দ্বারা সংখ্যা ফরম্যাট করা যাবে:
Dim TestStr as String
TestStr = Format(54590125.4, "###,##,##0.00") 'Results 54,590,125.40

গ. Math Functions - ম্যাথ ফাংশন্স - গাণিতিক ফাংশন্স। এসব ফাংশনের নামের পূর্বে Math. শব্দ লিখতে হবে।

১. Math.Abs -এ ফাংশন যে কোনো সংখ্যাকে পজিটিভ সংখ্যায় পরিণত করে। যেমন:
Math.Abs(-123) -returns 123
২. Math.Acos -এটা কোণের Cosine সংখ্যা থেকে ডিগ্রী রেডিয়ান দেবে। যেমন:
Math.Acos(0.9568) -returns 0.295 radians
৩. Math.Asin -এটা কোণের Sine সংখ্যা থেকে ডিগ্রী রেডিয়ান দেবে। যেমন:
Math.Asin(0.145268) -returns 0.14578 radians
অন্যান্য হলো: Atan [Arc Tan], Cos [Cosine], Cosh [Hyperbolic Cosine], Sin [Sine], Sinh [Hyperbolic Sine], Tan [Tangent], Tanh [Hyperbolic Tan], Exp [Exponential -e^y to the power of y], Pow [x^y Value], Log [Logarithm -base 2], Log10 [Logarithm -base 10], Sqrt [Square Root], Sign [Integer value with + or - Sign], Min [Smaller of two numbers], Max [Larger of two numbers]
আরো কিছু ম্যাথ ফাংশন হলো:
(ক) `Math.BigMul(1256, 2154) -returns 2705424`
(খ) `Math.IEEERemainder(45,7) -returns (45/7) reminder 3`
(গ) `Math.Round(45.8) -returns 46; Math.Round(45.81255, 3) -returns 45.823`
(ঘ) `Math.Truncate(45.81255) -returns 45`

ঘ. Type Conversion Functions -টাইপ (ভেরিয়েবল ধরন) রূপান্তর ফাংশন
১. Integer > < Boolean - উদাহরণ:
```
Dim a, b, c As Integer
Dim check As Boolean
a = 5 : b = 5 ' The following line of code sets check to True.
check = CBool(a = b) 'check is True
c = 0 ' The following line of code sets check to False.
check = CBool(c) 'check is False
```

২. Double > < Byte - উদাহরণ:
```
Dim aDouble As Double : Dim aByte As Byte
aDouble = 125.5678 ' The following line of code sets aByte to 126.
aByte = CByte(aDouble)
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৩. String > < Char, CChar - উদাহরণ:

```
Dim aString As String
Dim aChar As Char
aString = "BCD" ' CChar converts only the first character of the string.
aChar = CChar(aString) 'sets aChar to "B".
```

৪. String > < Integer, Char - উদাহরণ:

```
Dim someDigits As String
Dim codePoint As Integer
Dim thisChar As Char
someDigits = InputBox("Enter code point of character:") 'ask the user
codePoint = CInt(someDigits) 'convert it to integer and equate ti declared integer type
' The following line of code sets thisChar to the Char value of codePoint.
thisChar = ChrW(codePoint)
```

৫. String > < To Date and Time -উদাহরণ:

```
Dim aDateString, aTimeString As String
Dim aDate, aTime As Date
aDateString = "February 12, 1969"
aTimeString = "4:35:47 PM" ' The following line of code sets aDate to a Date value.
aDate = CDate(aDateString) ' The following line of code sets aTime to Date value.
aTime = CDate(aTimeString)
```

৬. Double > < To Integer -উদাহরণ:

```
Dim aDbl As Double
Dim anInt As Integer
aDbl = 2345.5678 ' The following line of code sets anInt to 2346.
anInt = CInt(aDbl)
```

৭. Double > < To Object -ভিজুয়্যাল বেসিক.নেটে Object -একটি টাইপ ভেরিয়েবল। উদাহরণ:

```
Dim aDouble As Double
Dim anObject As Object
aDouble = 2.7182818284 ' The following line of code sets anObject to a pointer to
aDouble.
anObject = CObj(aDouble)
```

দ্র: উপরের কোড অংশের কমেন্টে pointer -শব্দটি ব্যবহার করেছি। pointer কি? প্রোগ্রামিং ভাষায় pointer হলো: প্রোগ্রামিং ভাষার বস্তু [Object], যার মান [Value] ইঙ্গিতে নির্দেশ করে [Points to] কম্পিউটারের অন্যত্র মেমোরি ঠিকানায় সংরক্ষিত মানের দিকে। আরো বুঝতে আমরা গ্রন্থের 'সূচিপত্রের' কথা ভাবতে পারি। সূচিতে সংশ্লিষ্ট বিষয়ের উপর আলোচনার পৃষ্ঠা সংখ্যা থাকে। অন্যকথায়, বইয়ের সূচিতে দেওয়া এ পৃষ্ঠা নাম্বার হলো একটি পয়েন্টার যা আমাদেরকে জানিয়ে দেয় কোন্ পৃষ্ঠায় তথ্যাদি সংরক্ষিত [লিখিত] হয়েছে। পয়েন্টার ব্যবহারের মাধ্যমে কম্পিউটিং গতি বৃদ্ধি পায়। কারণ, একই ঠিকানায় সংরক্ষিত তথ্য বিভিন্ন কোড লাইনে ব্যবহৃত হয়ে থাকে।

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: ***Ready-made functions & Methods***

৮. String > < Char, CChar - উদাহরণ:

Dim aString As String
Dim aChar As Char
aString = "BCD" ' CChar converts only the first character of the string.
aChar = CChar(aString) 'sets aChar to "B".

৮. Double > < Byte, Short - উদাহরণ:

Dim aDouble As Double
Dim anSByte As SByte
aDouble = 39.501 ' The following line of code sets anSByte to 40.
anSByte = CSByte(aDouble)
Dim aByte As Byte
Dim aShort As Short
aByte = 100 ' The following line of code sets aShort to 100.
aShort = CShort(aByte)

> Byte data type -কম্পিউটারের মেমোরিতে সংরক্ষিত সর্বাধিক স্বল্পমাত্রার সাংখ্যিক তথ্যকে বলে বিট [Bit]। একেক বিটকে বিজ্ঞানীরা দু'টি মাত্র সাংখ্যিক কোড দ্বারা উপস্থাপন করেন। এগুলো হলো: ০ এবং ১। এ দু'টোর সমন্বয়ে ঘটিত ৮টি বিট হলো একেকটি বাইট [Byte]। সুতরাং ১ বাইট মানে ৮ বিট তথ্য।

৯. Double > < String - উদাহরণ:

Dim aDouble As Double
Dim aString As String
aDouble = 437.324 ' The following line of code sets aString to "437.324".
aString = CStr(aDouble)

১০. Double > < String - উদাহরণ:

Dim testNumber As Long = 1000 ' The following line of code sets testNewType to 1000.0.
Dim testNewType As Single = CType(testNumber, Single)

৬. Properties -ভিজুয়্যাল বেসিকে প্রোপার্টি

১. DateString -কম্পিউটারের সেটিং মুতাবিক বর্তমান তারিখ ও সময়। ব্যবহার:

MsgBox("The current date is " & DateString) 'returns 08-11-2016
MsgBox("The formatted date is " & Format(Now, "dddd, d MMM yyyy")) 'returns Thursday, 11 August 2016.

২. Now -এখনকার তারিখ ও সময়। ব্যবহার:

Dim ThisMoment As Date ' The following statement calls the Get procedure of the Visual Basic Now property.
ThisMoment = Now : MsgBox(ThisMoment) 'retuns 8/11/2016 12:44:03 PM

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৩. ScriptEngine -বর্তমান রান-টাইমের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngine) 'retuns- VB

৪. ScriptEngineBuildVersion -বর্তমান রান-টাইমের বিল্ড ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineBuildVersion) 'retuns- 50709

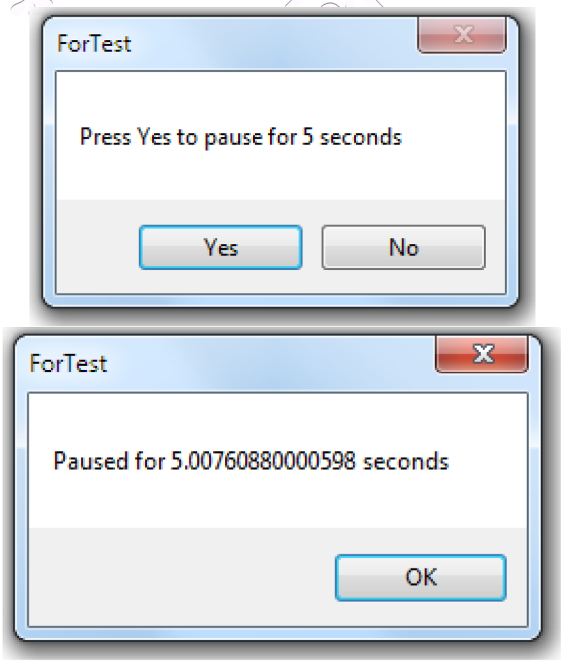
৫. ScriptEngineMajorVersion -বর্তমান রান-টাইমের মেজর ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineMajorVersion) 'returns 11

৬. ScriptEngineMinorVersion -বর্তমান রান-টাইমের মাইনোর ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineMinorVersion) 'returns 0

৭. TimeOfDay -দিনের বর্তমান সময়ের ওপর তারিখ ভ্যালু সেট বা রিটার্ন করে। উদাহরণ:
Dim currentTime As Date
currentTime = TimeOfDay 'returns 1:10:19 PM

৭. Timer -আগের মধ্যরাত থেকে বর্তমান পর্যন্ত কতো সেকেন্ড অতিবাহিত হয়েছে তার একটি মাত্রা। উদাহরণ:

```
Public Sub waitFiveSeconds()
    If TimeOfDay >= #11:59:55 PM# Then
      MsgBox("The current time is within 5 seconds of midnight" &
        vbCrLf & "The timer returns to 0.0 at midnight")
      Return
    End If
    Dim start, finish, totalTime As Double
    If (MsgBox("Press Yes to pause for 5 seconds", MsgBoxStyle.YesNo)) = MsgBoxResult.Yes Then
      start = Microsoft.VisualBasic.DateAndTime.Timer
      ' Set end time for 5-second duration.
      finish = start + 5.0
      Do While Microsoft.VisualBasic.DateAndTime.Timer < finish
        ' Do other processing while waiting for 5 seconds to elapse.
      Loop
      totalTime = Microsoft.VisualBasic.DateAndTime.Timer - start
      MsgBox("Paused for " & totalTime & " seconds")
    End If
End Sub
```



উপরের প্রোগ্রাম রানিংয়ের চিত্র।

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

শিক্ষার্থীরা অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং [OOP] কথাটি ইতোমধ্যে অন্যত্র দেখেছেন। এ সম্পর্কে বিস্তারিত আলোচনা জরুরী মনে করে এ পরিচ্ছেদ লেখার প্রয়াস। কম্পিউটার সফ্টওয়্যার সৃষ্টির জগতে OOP একটি আধুনিক ধারণা [concept]। ভিজুয়্যাল বেসিক.নেট এর পূর্বসুরী ছিলো শুধু ভিজুয়্যাল বেসিক। ভার্শন ৬.০ এর পর মাইক্রসফ্ট ভিজুয়্যাল বেসিক আর ডেভোলাপ করে নি। ভিজুয়্যাল বেসিক.নেট এর ওপরই মনোযোগ দিয়েছে। এর একটি অন্যতম কারণ, আধুনিক অন্যান্য ভাষা [যেমন c++, Java] হলো পূর্ণাঙ্গ OOP-নিয়ন্ত্রিত। মাইক্রসফ্ট ভিজুয়্যাল বেসিকের মতো নতুন একটি ভাষা তৈরি করতে সচেষ্ট হয় পৃথিবীব্যাপী অসংখ্য ভক্ত-অনুরক্ত প্রোগ্রামারদের সুবিধার্থে। সবাই চাচ্ছিলো একটি উন্নতমানের ভাষা যাতে সরলতা থাকবে- যেরূপ মূল ভিজুয়্যাল বেসিকে ছিলো, এসাথে আধুনিক অন্যান্য ভাষার সঙ্গে সামঞ্জস্যশীলতা নিশ্চিত করা। আর এরই ফলাফল স্বরূপ আমরা পেয়েছি পূর্ণাঙ্গ OOP-সহ মাইক্রসফটের এই .নেট ভাষাটি। এই ভাষার আরেক বৈশিষ্ট্য হলো উইন্ডোজ অপারেটিং সিস্টেমের ফাংশন, মেথড ইত্যাদি সরাসরি কাজে লাগানোর সুযোগ। এ সুযোগের নামই হলো API- Application Program Interface। মূলত একটি কোডিং সিস্টেম যেখানে সফটওয়্যার তৈরিকারী নিজের কোড দ্বারা সিস্টেমের কোডের সঙ্গে যোগাযোগ সৃষ্টি করতে পারেন।
যাক, ফিরে আসি OOP প্রোগামিং কনসেপ্টে। আমাদের এ জগতটি মূলত বস্তু বা অবজেক্টের তৈরি। একটি বাড়ি। এটা সার্বিকভাবে একটি বস্তু যা সৃষ্টি হয়েছে একাধিক বিভিন্ন অংশের সমন্বয়ে। যেমন: ইট, পাথর, বালি, দেওয়াল, ফ্লোর, কক্ষ, ছাদ, পিলার ইত্যাদি। এ-সবই বাড়ির অংশ বা প্রোপার্টি। অনুরূপ একটি বৃক্ষের মধ্যে বেশ ক'টি প্রোপার্টি চিহ্নিত করা যায়। যেমন: এর মূল, কাণ্ড, শাখা, প্রশাখা, পত্র, উচ্চতা, কাণ্ডের ব্যাস ইত্যাদি। কম্পিউটার বিজ্ঞানীরা এরূপ প্রাত্যহিক বস্তু ও এদের প্রোপার্টি গবেষণা করে OOP সৃষ্টি করেছেন প্রোগ্রামিং সুবিধার্থে।

যে কোনো প্রোগ্রামিং ভাষাকে OOP হিসেবে স্বীকৃতি পেতে হলে তার মধ্যে তিনটি মৌলিক জিনিস থাকতে হবে:

ক. **Encapsulation** -ঠিক যেরূপ ক্যাপসুলের ভেতর কিছু থাকে সেরূপ কোডগুলো বন্ধ অবস্থায় থাকা। অন্যকথায় বাইরের জগত থেকে গোপন রাখা।

খ. **Inheritance** -ঠিক যেরূপ কোনো প্রাণী নিজের বংশবিস্তার লাভে সমর্থ, সেরূপ নিজেকে কপি করার ক্ষমতা থাকা। অর্থাৎ উত্তরাধিকার ব্যবস্থা থাকতে হবে।

গ. **Polymorphism** -ঠিক একই দলভুক্ত বিভিন্ন বস্তুর মতো থাকতে হবে, একই দলভুক্ত একাধিক ভিন্ন বস্তু।

এই তিনটি বিষয়ের ওপর আরো কিছু ব্যাখ্যা তুলে ধরার প্রয়াস পাচ্ছি। এগুলো বুঝে এসে গেলে OOP সম্পর্কে ধারণাও আরো স্পষ্ট হয়ে ওঠবে।

ক. Encapsulation

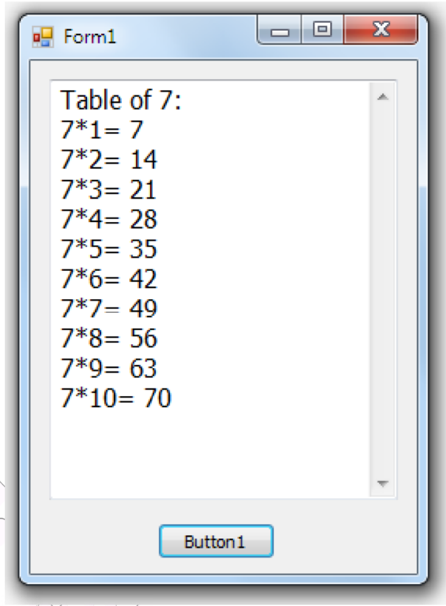
কোডিংয়ের ক্ষেত্রে বার বার ব্যবহারযোগ্য এবং ভিন্ন এ্যাপলিকেশনে বহনযোগ্য কিছু ইউনিট তৈরি করা যায়। এগুলোকে বলে আত্মনির্ভরশীল মডিউল [self-contained module]। এসব মডিউলে থাকে তথ্য প্রসেস করার বিভিন্ন ফাংশন। ব্যবহারকারী কর্তৃক নিরূপণকৃত তথ্য-ধরনকে [user-defined data type] বলে ক্লাস [class]। প্রত্যেক ক্লাসে থাকে তথ্য [data]। আরো থাকে এক বা একাধিক পদ্ধতি [method] যাদের মাধ্যমে তথ্যাদি রদবদল করা যায়। ক্যাপসুলের মতো আত্মনির্ভরশীল আবদ্ধ ক্লাসের ডাটা অংশকে বলে সময়ের পরিবর্তনশীল তথ্য বা ইন্সট্যান্ট ভেরিয়েবল্স [instant variables]। আর ক্লাসের একটি ইন্সট্যান্ট হলো একটি বস্তু [object]। আমরা এগুলোকে পাবলিক লাইব্রেরি সিস্টেমের সঙ্গে তুলনা করতে পারি। ক্লাস হতে পারে লাইব্রেরির সদস্য এবং আমির ও সাকিয়া হতে পারে এ লাইব্রেরি ক্লাসের দু'টি ইন্সট্যান্ট [দু'টি বস্তু]। আশারাখি এ কথাগুলো পাঠ করে আপনি Encapsulation সম্পর্কে আরো কিছুটা স্পষ্ট ধারণা পেয়ে গেছেন। তবে বাস্তব ক্ষেত্রে একটি কোড উদাহরণ দেখুন। এতে বিষয়টি আরো স্পষ্ট হবে।

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

ভিজুয়্যাল বেসিক.নেটে **Encapsulation** ব্যবহারের উদাহরণ: একটি গুণের নামতা:

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    TextBox1.Text = "" 'clear text
    MySub() 'calling another sub
   End Sub
    'called from button clicked event sub
   Public Sub MySub()
      Dim obj As ClassA = New ClassA(7) 'create object of ClassA
      obj.table() 'calling member method by object [function call which is in classA]]
   End Sub
End Class
====
'class declaration
Public Class ClassA 'ClassA starts
   Private Value As Integer 'private member variable
   Sub New(V As Integer) 'constructor with 1 argument
      Value = V
   End Sub
   Public Function table() 'member method to show member variable
      Dim Heading, TableV As String
      Heading = "Table of " & Value & ": " & vbCrLf 'heading text String with
return vbCrLf Character
      Form1.TextBox1.Text += Heading 'display heading in the TextBox
      For i As Integer = 1 To 10 Step 1 'calculate times table and display
         TableV = Value & "*" & i & "= " & Value * i & vbCrLf 'calculate and
put into variable
         Form1.TextBox1.Text += TableV 'display '+=' means add text to
existing text.
      Next
      Return Nothing
   End Function
End Class 'classA ends
```



বায়ের প্রোগ্রামের ফলাফল।

খ. **Inheritance**

ক্লাসগুলো সৃষ্টি হয় হাইআর্কির [hierarchies] সূত্র ধরে। অর্থাৎ ক্রমাধিকারের মাধ্যমে এগুলো অস্তিত্বশীল হয়। ইনহেরিট্যান্স [Inheritance] বা উত্তরাধিকার থাকায় কোড কাঠামো ও মেথড একটি ক্লাস থেকে হাইয়ার্কির নিচ দিকে যেতে সক্ষম হয়। এর ফলে জটিল সিস্টেমেও কম প্রোগ্রামিংয়ের জরুরত হবে। অন্যকথায় ভিন্ন কোনো পদক্ষেপ হাইয়ার্কির নিম্নতম স্তরে যুক্ত করতে হলে পুরো হাইয়ার্কিকে অপরিবর্তিত রেখেই শুধুমাত্র ঐ পদক্ষেপটি তৈরি করে যুক্ত করা সম্ভব হবে। এই পদক্ষেপের বাকি সকছিুই উত্তরাধিকারসূত্রে পাওয়া যাবে [inherited]। এভাবে প্রস্তুতকৃত বস্তু পুনঃব্যবহারের সুযোগ থাকাটাকে অবজেক্ট প্রযুক্তির ক্ষেত্রে সর্বাধিক বড় সুবিধা হিসাবে ধারণা করা হয়।

একটি ক্লাস যখন অপর আরেকটি ক্লাস Inherit করে বা উত্তরাধিকারসূত্রে প্রাপ্ত হয় তখন একে বলে derived [উদ্ভূত] ক্লাস। অপরদিকে উদ্ভূত ক্লাস থেকে যে ক্লাস উত্তরাধিারসূত্রে প্রাপ্ত হয় তাকে বলে Base Class [মূল ক্লাস]। কেউ কেউ অন্যান্য নামও ব্যবহার করেন, যেমন: প্যারেন্ট [অভিভাবক], চাইল্ড [সন্তান] ইত্যাদি। এবার একটি উদাহরণ দেখুন।

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        TextBox1.Text = "" 'clear text
        InVokes()
    End Sub
```

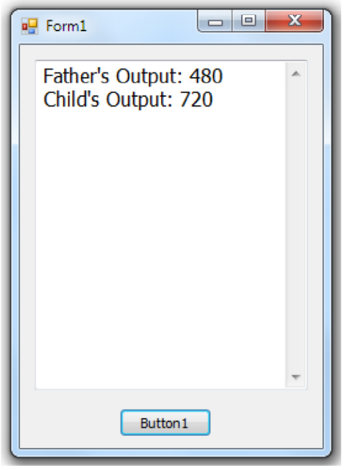
১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

```
    Sub InVokes() 'method- called from button's click event sub
            Dim obj As Child = New Child(5, 48)
            TextBox1.Text += "Father's Output: " & obj.FatherOutput & vbCrLf
            TextBox1.Text += "Child's Output: " & obj.ChildOutput
    End Sub
End Class
'=======Base Class=======
Public Class Father 'start of base class Father
   Protected Resources As Integer
   Protected WorkInHours As Integer
   Public Sub New(r As Integer, w As Integer) 'constructor with 2 arguments
      Resources = r
      WorkInHours = w
   End Sub

   Public Function FatherOutput() As Integer 'method
      Dim Output As Integer = Resources * WorkInHours
      Return Output
   End Function
End Class
'========Inherited Class=======
Public Class Child : Inherits Father 'derived class
   Private Resources As Integer
   Private WorkInHours As Integer
   Public Sub New(r As Integer, w As Integer) 'constructor with 2 arguments
      MyBase.New(10, 48)
      Me.Resources = r
      Me.WorkInHours = w
   End Sub

   Public Function ChildOutput() As Integer 'method
      Dim Output As Integer = (Me.Resources + MyBase.Resources) * WorkInHours
      Return Output
   End Function
End Class
```



বায়ের প্রোগ্রাম রানের ফলাফল

খ. Polymorphism

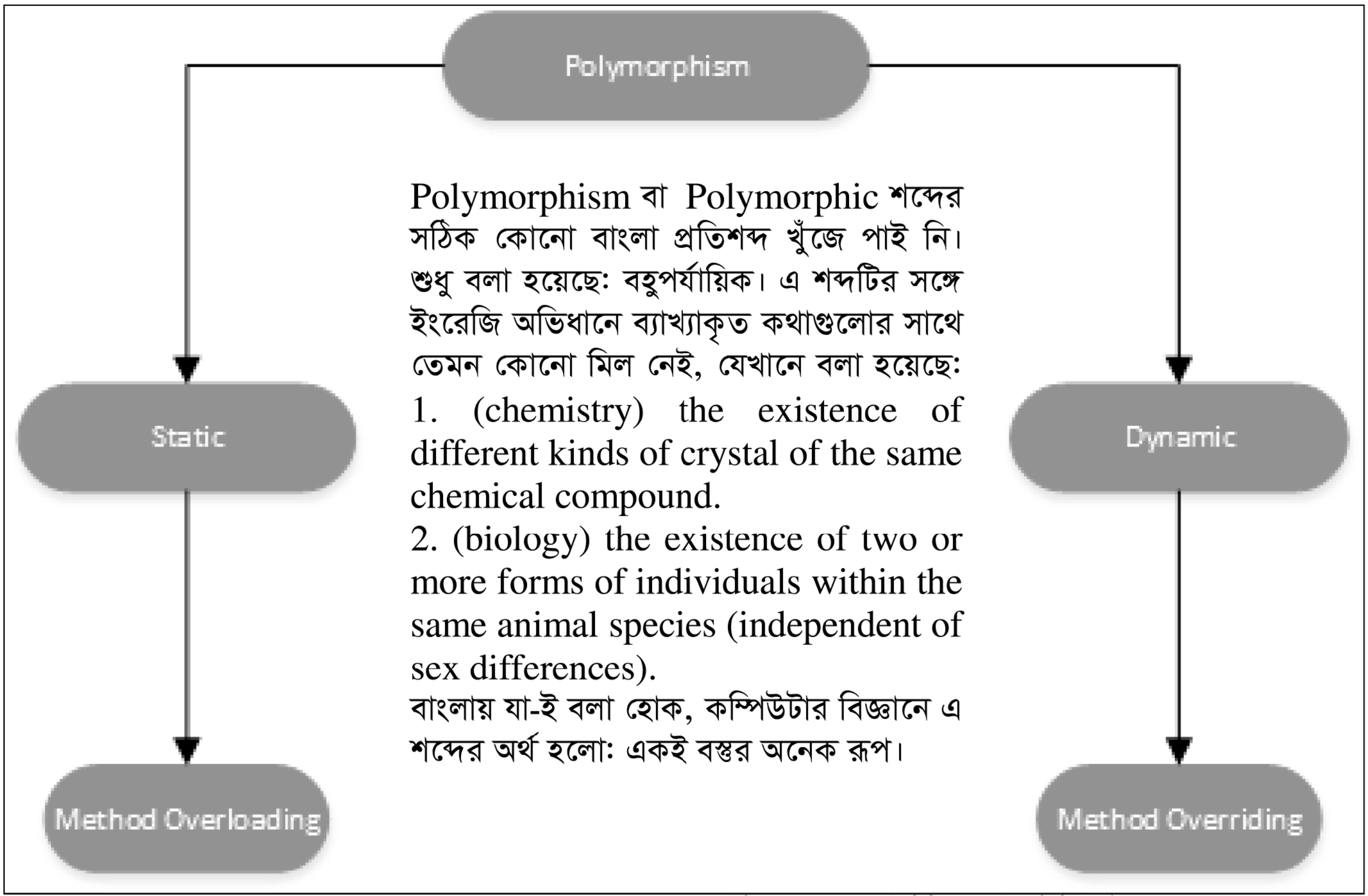
শব্দটি আসলে রসায়ন ও জীববিদ্যা থেকে ধার-করা। রসায়ন শাস্ত্রে Polymorphism-এর অর্থ হলো একই রসায়নভুক্ত কম্পাউন্ড [যুক্ত পদার্থ] দ্বারা সৃষ্ট বিভিন্ন ক্রিস্ট্যাল [স্ফটিক]। জীববিদ্যায় এ শব্দের অর্থ হলো একই প্রজাতিভুক্ত বিভিন্ন প্রাণীর অস্তিত্ব যা মূলত লিঙ্গ নির্ভরশীলহীন।

অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়ে বিমূর্ত [abstract] কোড কাঠামো ও মেথড [পদ্ধতি] তৈরি সম্ভব যাদের যথার্থ ধরন বা টাইপ রানটাইম [প্রোগ্রাম চলাকালীন অবস্থা] এর পূর্বে জানা যায় না। যেমন ধরুন, স্ক্রীনের কারসার মাউস চলাকালে তার আকার তীর [এ্যারো] থেকে বদলে রেখায় [লাইন] পরিণত হতে পারে। এটা নির্ভর করবে প্রোগ্রামের ভাবধারার ওপর। সুতরাং আপনি প্রোগ্রামার হিসেবে স্ক্রীনে মাউস চালানোর জন্য লিখবেন 'cursor' রুটিন [কোড বল্ক]। আর পলিমোরফিজম রানটাইমের সময় মাউসের আকার প্রয়োজনমাফিক রদবদল করবে। এটা নতুন এই আকারকে সহজে সম্পৃক্ত করতেও সুযোগ সৃষ্টি করে দেবে। অন্যকথায় রসায়ন শাস্ত্রে যেরূপ একই রসায়নভুক্ত কম্পাউন্ড থেকে সৃষ্টি হতে পারে বিভিন্ন ক্রিস্ট্যাল এবং জীববিদ্যায় একই প্রজাতিভুক্ত বিভিন্ন প্রাণীর অস্তিত্ব সম্ভব- তদ্রূপ অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়েও রান-টাইমের সময় একটি অবজেক্ট প্রয়োজনমাফিক তার আকার, আয়তন, কর্ম ইত্যাদিও রদবদল করার ক্ষমতা রাখতে পারে। আর একেই বলে **Polymorphism**। কোডে পলিমোরফিজম্ হয় মেথড অভাররাইডিং ও মেথড অভারলোডিং এর মাধ্যমে। উদাহরণ দ্বারা বুঝা অনেকটা সহজ। [পরের পৃষ্ঠায় দেখুন]

১০ম পরিচ্ছেদ: ***Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং***

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

Polymorphism-এর দু'টি রূপ আছে: ১. Static Polymorphism [স্থিত পলিমোরফিজম্] ও ২. Dynamic Polymorphism [বেগবান পলিমোরফিজম্]। নিচের ব্লক চিত্র থেকে উভয়টির স্বরূপ সম্পর্কে অতিরিক্ত ধারণা পাওয়া যেতে পারে।



উপরের ডায়াগ্রাম থেকে স্পষ্ট, পরিমোরফিজমের স্টেটিক [স্থিত] শাখার মেথড [কোড পদ্ধতি] হলো অভারলোডিং [অতিরিক্ত বোঝাই করা]। অরপদিকে ডাইনামিক [বেগবান] শাখার মেথড হবে অভাররাইডিং [পদদলিতকরণ!]।

১. Static Polymorphism [স্থিত পলিমোরফিজম্]: একটি উদাহরণ

একে প্রাথমিক বন্ধন বা Early Binding কিংবা ভাষা পরিবর্তনকাল অর্থাৎ Compile Time Polymorphism-ও বলে। অন্যকথায় সিদ্ধান্ত নেওয়া হয় কম্পাইল টাইমে। মেথড অভারলোডিং হলো কম্পাইল টাইমেই কম্পাইলার জানতে পারে কোন্ মেথড সম্পাদিত হবে। কম্পাইল টাইমে কম্পাইলার প্যারামিটার সংখ্যা এবং প্যারামিটারের ধরন বুঝে সিদ্ধান্ত নেয় কোন্ অভারলোডেড মেথড সম্পাদিত [execute] হবে। উদাহরণ কোডটি দেখে নিই।

```
Public Class Student 'class starts
   'show method with 3 overloaded versions
   Public Overloads Function show(name As String) As String 'method with 1 parameter
      Return name
   End Function
```

[বাকি কোড পরের পৃষ্ঠায়]

```
    Public Overloads Function show(name As String, rollNo As String) As String 'method with 2 parameter
        Return (name + " " + rollNo)
    End Function

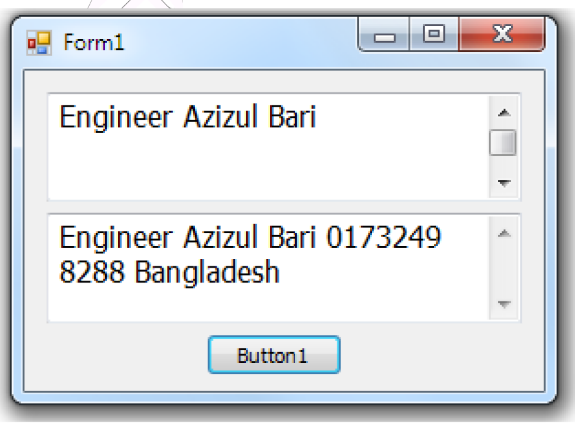
    'method with 3 parameter
    Public Overloads Function show(name As String, rollNo As String, Section As String) As String
        Dim St As String = (name + " " + rollNo + " " + Section)
        Form1.TextBox2.Text = St 'display result
        Return Nothing
    End Function
End Class

Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        TextBox1.Text = "" 'clear text
        TextBox2.Text = "" 'clear text
        CallSub()
    End Sub

    'called from button clicked event sub
    Public Sub CallSub()
        Dim s As Student = New Student()
        TextBox1.Text += (s.show("Engineer Azizul Bari")) & VbCrlf 'call show method with 1 param
        'call show method with 3 param
        TextBox1.Text += (s.show("Engineer Azizul Bari", "0173249 8288", "Bangladesh")) & VbCrlf
    End Sub
End Class
```

২. Dynamic Polymorphism [বেগবান পলিমোরফিজম]: একটি উদাহরণ

একে রান-টাইম (চলাকালীন) বা লেইট-বাইন্ডিং (দেরিতে-চলন্ত) পলিমোরফিজম-ও বলে। মেথড অভাররাইডিং এর মাধ্যমে এটা সম্পাদিত হয়। মেথড অভাররাইডিং মানে একই মেথডের বিভিন্ন রূপকে কল করা। কোন্ রূপকে কল করা হবে তা নির্ভর করবে প্রোগ্রাম চলাকালে বেগবান অবস্থায়। অন্যকথায়, রান-টাইম পলিমোরফিজমে কোন্ মেথড কল করা হবে তা নির্দিষ্ট হয় রেফারেন্স ভেরিয়েবল কর্তৃক পছন্দকৃত বস্তু থেকে। উদারহণ থেকে এসব কথার সঠিক অর্থ হয়তো অনুধাবন হতে পারে।



উপরের প্রোগ্রাম পরিচালনার ফলাফল

```
Public Class Add 'base class starts
    'Virtual method of Base Class
    Public Overridable Function Calculate(num1 As Integer, num2 As Integer) As Integer
        Form1.TextBox1.Text = ("Virtual method of base class!")
        Dim result As Integer = 0
        result = num1 + num2
        Return result
    End Function
End Class 'Base Class Ends
```

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: ***Object Oriented Programming [OOP]***

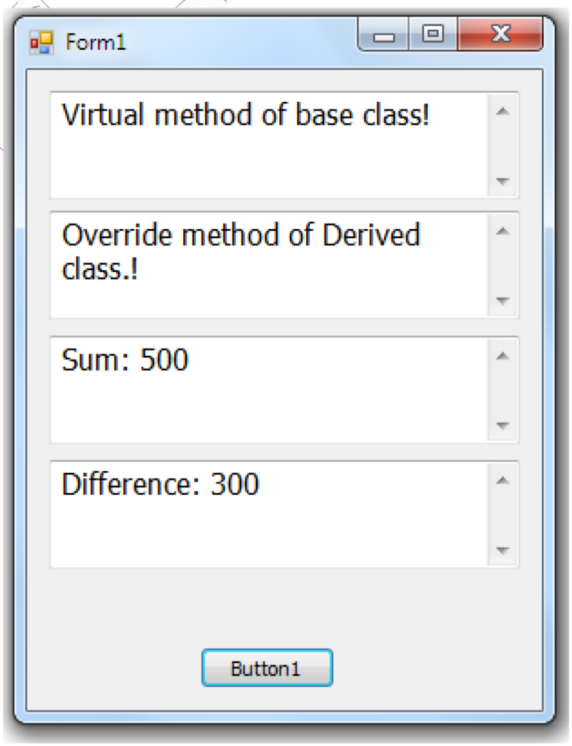
```
Public Class Subtract : Inherits Add 'Derived class starts
    'override method from Base class
   Public Overrides Function Calculate(num1 As Integer, num2 As Integer) As Integer
      Form1.TextBox2.Text = ("Override method of Derived class.!")
      Dim result As Integer = 0
      result = num1 - num2
      Return result
   End Function
End Class 'Derived class ends

Public Class Form1

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      AddMinus()
   End Sub

   'called from button clicked event sub
   Public Sub AddMinus()
      Dim a1 As Add = New Add()
      Dim AddRes As Double = (a1.Calculate(400, 100))
      TextBox4.Text = "Sum: " & AddRes
      Dim a2 As Add = New Subtract() ' instance of the derived class using base class reference
      Dim Dif As Double = (a2.Calculate(400, 100))
      TextBox3.Text = "Difference: " & Dif
   End Sub
End Class
```

দ্রষ্টব্য: এ পরিচ্ছেদে উপস্থাপিত উদাহরণগুলো মূলত অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়ের স্বরূপ বুঝানোর জন্য দেওয়া হয়েছে। আসলে যেসব ফলাফল এগুলো থেকে পাওয়া যাচ্ছে তা অন্যভাবে অনেকটা সজহ উপায়ে প্রোগ্রাম করার মাধ্যমে অর্জন সম্ভব। আর বাস্তব ক্ষেত্রে এরূপ সাধারণ প্রোগ্রামের জন্য ব্যবহার মোটেই প্রয়োজনীয়ও নয়। যাক, আশাকরি পাঠকরা OOP-এর সাথে সম্পৃক্ত তিনটি মৌলিক বিষয় তথা, **Encapsulation, Inheritance** এবং **Polymorphism** সম্পর্কে অত্যল্প হলেও ধারণা পেয়ে গেছেন। আর এটুকু অর্জনই এ অধ্যায় রচনার উদ্দেশ্য ছিলো। আমরা OOP-নিয়ে খুব একটা মাথা ঘামাবো না। পরবর্তীতে যেসব প্রোগ্রাম উদাহরণ উপস্থাপন করবো তা হবে ভিজুয়্যাল বেসিক.নেট প্রোগ্রাম করার কৌশল বুঝা, কোড জানা এবং অভিজ্ঞতা অর্জন করার নিমিত্তে। আমরা কিন্তু OOP-ই করবো কিন্তু তা হবে আমাদের অজান্তে!



উপরের প্রোগ্রাম পরিচালনার ফলাফল

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

ভিজুয়্যাল বেসিক.নেট দ্বারা ফাইল নিয়ন্ত্রণ: VB.Net File Handling

প্রোগ্রামিং এর একটি গুরুত্বপূর্ণ অংশ হলো তথ্য বা ডাটা ফাইলে লেখা, সংরক্ষণ ও প্রয়োজনে পাঠ করা। যে উপায়ে তা করতে হয় তাকে বলে File Streaming। এর অর্থ, কম্পিউটারের মেমোরিতে সংরক্ষিত ফাইল পড়া [Read] ও লেখা [Write]। প্রত্যেক প্রোগ্রামিং ভাষায় ফাইল নিয়ে কাজ করার কোড আছে। ভিজুয়্যাল বেসিক.নেটেও ফাইল নিয়ন্ত্রণের জন্য শক্তিশালী স্ট্রিমিং সিস্টেম তৈরি করা হয়েছে। এ পরিচ্ছেদে বিভিন্ন উদাহরণসহ ফাইল নিয়ে কিভাবে কাজ করতে হয় তার ওপর আলোচনা করবো।

File- সংজ্ঞা: নামকরণকৃত মেমোরি ব্লক বা বাক্স যাতে সংরক্ষিত থাকে তথ্যাদি। কম্পিউটারের ডিস্কে [হার্ড ডিস্ক, সিডি, ভিসিডি, পেন-ড্রাইভ] সকল ফাইল প্যাথ [path] দ্বারা সনাক্ত করা হয়।
যেমন: **C:\Users\htci\Desktop\Copied\Islamic History Project**

Streaming- সংজ্ঞা: প্যাথ [রাস্তা] ব্যবহার করে কোডের মাধ্যমে ফাইল খোলা, লেখা ও পাঠ করাকে বলে স্ট্রিমিং। খোলা ফাইলকে প্রোগ্রামিং ভাষায় স্ট্রিম **[Stream]** বলে।

ইনপুট/আউটপুট [Input/Output -I/O] ক্লাস

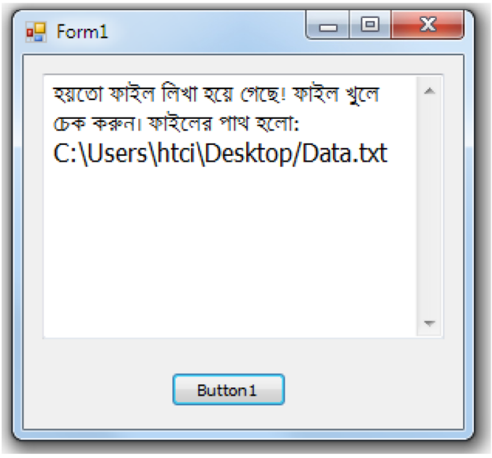
ভিজুয়্যাল বেসিক.নেটে ফাইল নিয়ন্ত্রণের জন্য আই/অ সুযোগ-সুবিধার মূলে আছে সিস্টেমের নামকৃত [namespace] ক্লাস যাকে System.IO বলে। অর্থাৎ System.IO থেকে সব আই/অ ক্লাস উত্তরাধিকারসূত্রে প্রাপ্ত। যেসব ক্লাস আমরা ব্যবহার করতে পারবো তার মধ্যে উল্লেখযোগ্য ক'টি হলো:
System.IO.Path, System.IO.StreamReader, System.IO.StreamWriter, System.IO.StringReader এবং System.IO.StringWriter। এ পরিচ্ছেদে বিভিন্ন উদাহরণে এগুলোর ব্যবহার ও ব্যাখ্যা আসবে।

ক. স্ট্রিমরাইটার ক্লাস [StreamWriter Class]: এই ক্লাস দ্বারা খোলা ফাইলে [অর্থাৎ স্ট্রিমে] ক্যারাক্টার লেখা হয়। নিচের উদাহরণ দ্বারা বিষয়টি বুঝে আসবে।

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      WriteToFile() 'sub call
   End Sub
 'called from button clicked event sub
   Public Sub WriteToFile()
      'creating instance of StreamWriter class, and opening specific file
      Dim FOut As System.IO.StreamWriter = New System.IO.StreamWriter("C:\Users\htci\Desktop/Data.txt")
       FOut.WriteLine("বিসমিল্লাহির রাহমানির রাহীম" & vbCrLf & "ভিজুয়্যাল বেসিক.নেট একটি সহজ প্রোগ্রামিং ভাষা, অথচ খুব শক্তিশালী। আপনি অতি সহজে এ ভাষা শিখে নিতে পারেন। প্রয়োজন শুধু শেখার স্পৃহা ও ধৈর্য, এই যা!")    'writing a string to file
      FOut.Close()   'closing file -don't forget that!
       TextBox3.Text = "হয়তো ফাইল লিখা হয়ে গেছে! ফাইল খুলে চেক করুন। ফাইলের পাথ হলো: " & vbCrLf     & "C:\Users\htci\Desktop/Data.txt"
    End Sub
End Class
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

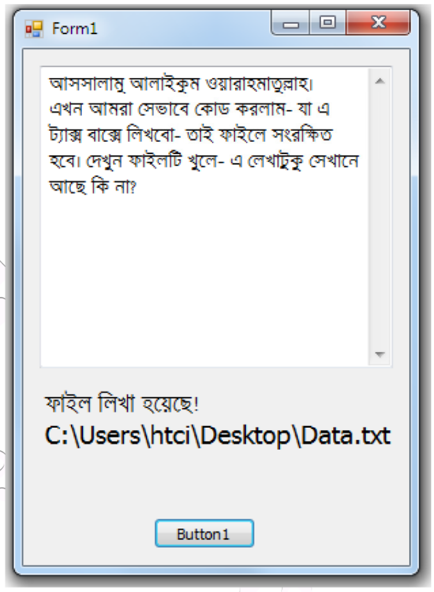


ডানের ছবিতে প্রোগ্রাম রান করার ফলাফল তুলে ধরেছি। লক্ষ করুন, বাংলা ইউনিকোড লেখা কিভাবে প্রদর্শিত হয়েছে। বিজয় কিবোর্ড বা অভ্র কিবোর্ড দিয়ে আপনি ইউনিকোড লিখতে পারেন। আরেকটি ব্যাপার এখানে উল্লেখ্য, কোডে ফাইলের প্যাথ দেওয়া আছে- কিন্তু, এই নামে প্যাথে কোনো ফাইল না থাকলে ভিজুয়্যাল বেসিক.নেট একটি ফাইল সৃষ্টি করবে। অবশ্য প্যাথের সবগুলো টেক্সট সঠিক হতে হবে।

লেখা সংরক্ষণ: আমরা চাইলে কোডটিতে সামান্য রদবদল এনে টেক্সটবাক্সে কিছু লিখে তা ফাইলে সংরক্ষণ করতে পারি। দেখুন নিচের পরিবর্তিত WriteToFile() সাবরুটিনটি।

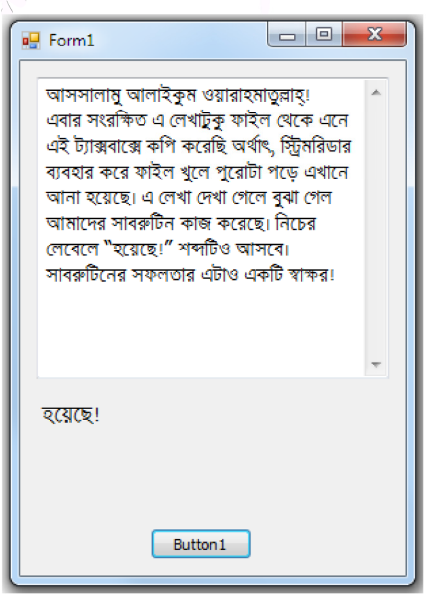
```
Public Sub WriteToFile()
      If TextBox3.Text <> "" Then 'if textBox3 dosen't have text in it nothing
will happen
            Dim WrittenText As String = TextBox3.Text 'put text into a variable
            'creating instance of StreamWriter class, and opening specific file
            Dim FOut As System.IO.StreamWriter = New
System.IO.StreamWriter("C:\Users\htci\Desktop\Data1.txt")
            FOut.WriteLine(WrittenText)
            FOut.Close()    'closing file
             Label1.Text = "ফাইল লিখা হয়েছে!" & vbCrLf &
"C:\Users\htci\Desktop\Data.txt"
      End If
    End Sub
```

এবার প্রোগ্রাম চালিয়ে টেক্সটবাক্সে কিছু লিখলে তা ফাইলে সংরক্ষিত হবে। দেখুন ডানের রান-টাইম চিত্রটি।



ক. স্ট্রিমরিডার ক্লাস [StreamReader Class]: এই ক্লাস দ্বারা খোলা ফাইল [অর্থাৎ স্ট্রিম থেকে] ক্যারাক্টার পাঠ করা হয়। নিচের উদাহরণ দ্বারা বিষয়টি বুঝে আসবে।

```
'called from button clicked event sub
    Private Sub ReadFile()
        'creating instance of StreamReader class, and opening specific file
        Dim FIn As System.IO.StreamReader = New System.IO.StreamReader
      ("C:\Users\htci\Desktop\Data1.txt")
        TextBox3.Text = FIn.ReadToEnd() 'read the whole file at once
        FIn.Close()   'closing file
         Label1.Text = "হয়েছে!"
End Sub
```



উপরের সাবরুটিন রান করার ফলে ফর্মের টেক্সটবাক্সে আসা ফাইলের লেখা ডানের ছবিতে চিত্রিত হয়েছে।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

Using the OpenFileDialog / SaveFileDialog Class: ফা্ইল খোলার ডায়লগের ব্যবহার

ফাইল নিয়ন্ত্রণ সম্পর্কে এ পর্যন্ত আমরা যা জেনেছি তাহলো স্ট্রিমরিডার ক্লাসের মাধ্যমে ফাইল খোলা, পড়া ও প্রদর্শন। আমরা এটাও জেনেছি কিভাবে ফাইলে লিখা যায়। এজন্য প্রয়োজন স্ট্রিমরাইটার ক্লাস। তবে উভয় ক্ষেত্রেই আমাদেরকে ফাইলের প্যাথ [path] পুরোটা লিখে দিতে হয়েছে। কিন্তু আমরা সচরাচর দেখে আসছি ওয়ার্ড প্রসেসর, পেইন্ট প্রোগ্রাম ইত্যাদি থেকে ফাইল খুলতে হলে একটি ডায়লগ আসে। আমরা যে ধরনের ফাইল খুলবো তা চয়েজ করে 'OK' বোতামে ক্লিক করলেই ফাইলটি খুলে টেক্সটবাক্সে বা ক্যানভাসে প্রদর্শিত হয়। অনুরূপ নতুন ফাইল সংরক্ষণকালেও ফা্ইল সংরক্ষণের ডায়লগ আসে। এ দু'টো কোডের মাধ্যমে কিভাবে নিয়ন্ত্রণ করা হয়? এখনই আমরা জেনে নেবো।

ভিজুয়্যাল বেসিক.নেটে ফাইল খোলা ও সংরক্ষণের দু'টি ক্লাস আছে। এদের নাম যথাক্রমে: OpenFileDialog() ও SaveFileDialog() । উদাহরণের মাধ্যমে এবার এ দু'টোর ব্যবহার উপস্থাপন করবো।

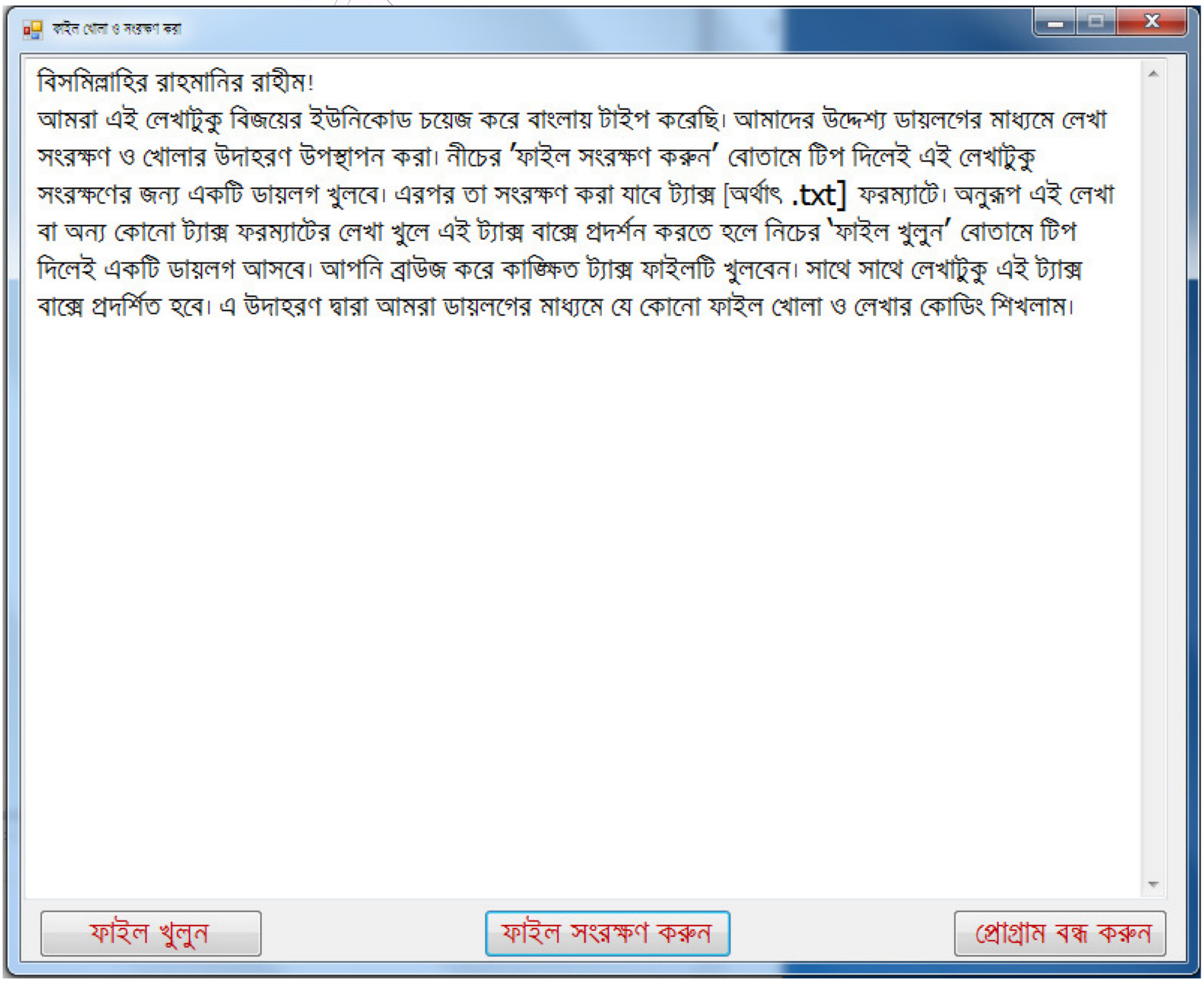
প্রথমে ফর্মের কোডটি দেখুন:

```
Public Class Form1
    'this sub clls the OpenFile Sub which in turn Opens the OpenDialog
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        OpenTheFile() 'sub call
    End Sub
    Sub OpenTheFile()   'this sub diplays open dialog
        Dim Open As New OpenFileDialog() 'new instance of file opening dialog class
        Dim myStreamReader As System.IO.StreamReader 'dimension a streamReader Object
        Open.Filter = "Text[*.txt*]|*.txt|All Files [*,*]|*.*" 'text files filters
        Open.CheckFileExists = True 'this function makes sure the file exists before opening
        Open.Title = "ফাইল খুলুন" 'open dialog title text'
        Open.ShowDialog(Me) 'shows the dialog
        Try 'Try is a debug keyword. If any problem it will catch it and jump to catch ex as exception line below
            Open.OpenFile() 'open chosen file
            myStreamReader = System.IO.File.OpenText(Open.FileName) 'assign file to StreamReader object
            TextBox1.Text = myStreamReader.ReadToEnd() 'read and display the whole file
        Catch ex As Exception 'any exception will be displayed
        End Try
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        If TextBox1.Text <> "" Then 'make sure there is text to be saved
            SaveTheFile() 'sub call
        End If
    End Sub
    Sub SaveTheFile() 'this dub displays Save Dialog
        Dim Save As New SaveFileDialog() 'new instance of file saving dialog class
        Dim myStreamWriter As System.IO.StreamWriter 'dimension a streamWriter Object
        Save.Filter = "Text (*.txt)|*.txt|All files(*.*)|*.*" 'text filters
        Save.CheckPathExists = True
        Save.Title = "ফাইল সংরক্ষণ করুন" 'saveDialog title text
        Save.ShowDialog(Me) 'Me means ShowDialog class itself
        Try
            myStreamWriter = System.IO.File.AppendText(Save.FileName) 'append already saved file- shows a replace
dialog
            myStreamWriter.Write(TextBox1.Text) 'write all text of the textBox to the file
            myStreamWriter.Flush() 'clear memory
        Catch ex As Exception
        End Try
    End Sub
   Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        Close() 'close form and exit application
    End Sub
End Class
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

পূর্বের পৃষ্ঠার কোড রানের ফলাফল নিচে দেখানো হয়েছে। আপনি চাইলে একটি নতুন প্রজেক্ট তৈরি করতে পারেন। লক্ষ করুন, ইউনিকোড দিয়ে কিভাবে বাংলা লিখেছি। এছাড়া ইউন্ডোজের বোতামগুলোর টেক্সট প্রোপার্টিও ইউনিকোডের মাধ্যমে বাংলায় করেছি। নিজের ভাষায় এভাবে ইউজার ইন্টারফেইস তৈরি করতে পেরে অনেকটা স্বস্তি ও আনন্দবোধ করছি। মাত্র কয়েক বছর পূর্বেও এভাবে করা কঠিন ছিলো। ডিজাইনার ফন্ট [যা মূলত ইংরেজি ক্যারাক্টার] দিয়ে অতীতে কাজ করতে হয়েছে। এখন সব ইউনিকোড প্রায় সকল উইন্ডোজ সিস্টেম সাপোর্ট করে। সুতরাং বাংলায় সফ্টওয়্যার তৈরি অনেকটা সহজ হয়েছে। আশা করছি ভবিষ্যতে বাংলা ভাষাভাষি প্রোগ্রামাররা এই সুযোগ ব্যবহার করবেন ও সবার জন্য তৈরি করবেন মনোমুগ্ধকর মাতৃভাষা-ভিত্তিক কম্পিউটার সফট্ওয়্যার।



চিত্র ১৯: আগের পৃষ্ঠার কোড রান করার ফলাফল। লক্ষ করুন: জানালার পুরোটাই বাংলায় উপস্থাপিত হয়েছে। এজন্য আমরা বাংলা ফন্ট চয়েজ করি নি। শুধুমাত্র বিজয়ের ইউনিকোড টাইপ সিস্টেম অন করেছি। মাইক্রসফটের ভিজুয়্যাল স্টুডিও এক্সপ্রেস ফর ডেস্কটপ ইউনিকোড সাপোর্ট করে। এ কারণেই বাংলায় সবকিছু সম্ভব।

ভিজুয়্যাল বেসিক.নেটে ফাইল ম্যানেজমেন্ট সম্পর্কে আরো এক'দুটো ব্যাপার জানার আছে। পরবর্তী পৃষ্ঠাসমূহে উদাহরণসহ এগুলো তুলে ধরছি।

১১তম পরিচ্ছেদ: *Working with Files*

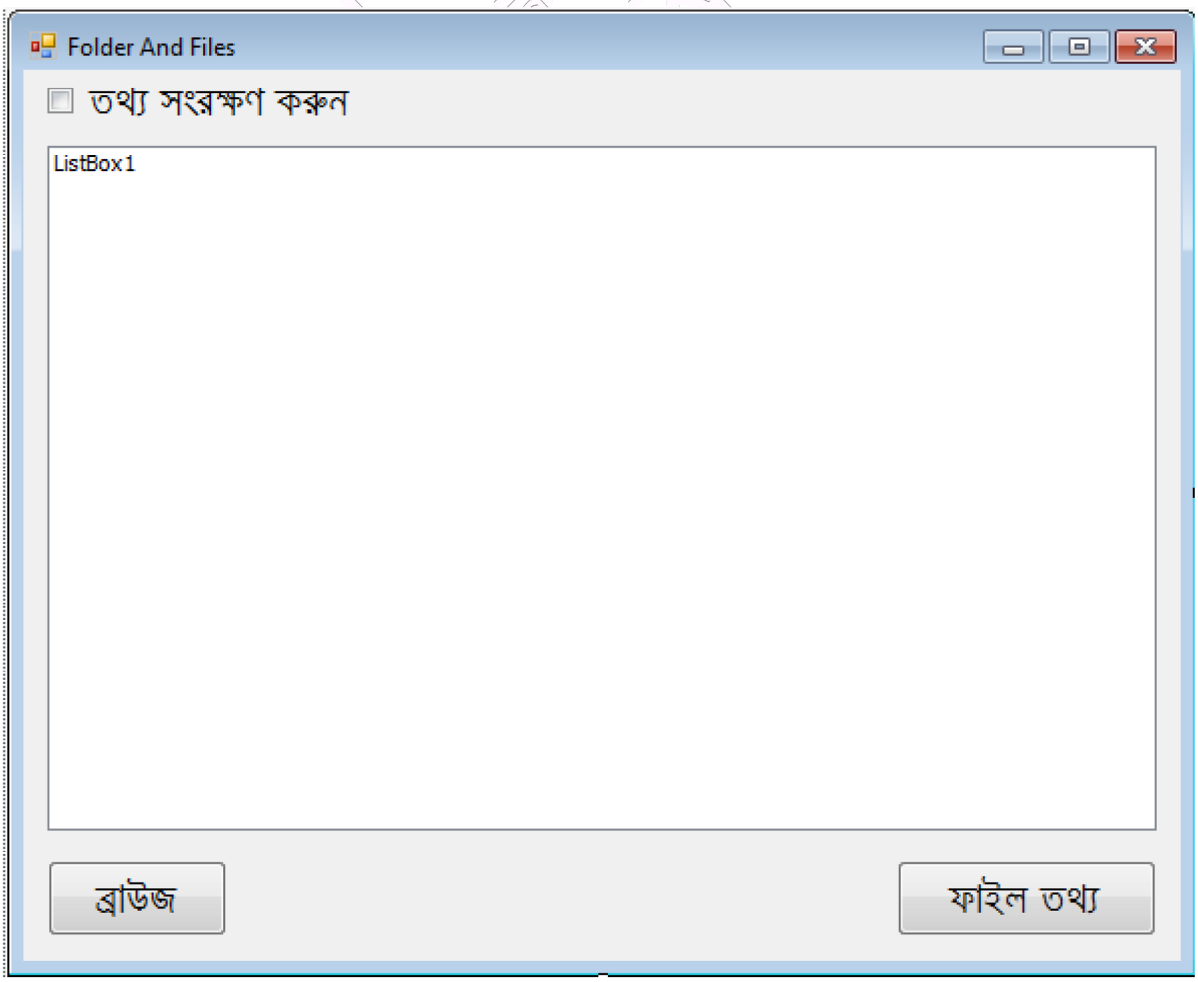
ফাইল নিয়ে কাজ : *Working with Files*

উদাহরণ-১: My.Computer.FileSystem Object ব্যবহার করে ফা্ইল নিয়ন্ত্রণ

আমরা ইতোমধ্যে ফা্ইল নিয়ন্ত্রণ সম্পর্কিত যেসব ব্যাখ্যা ও উদাহরণ তুলে ধরেছি তা সব্ই ছিলো **System.IO namespace** -এর অন্তর্ভুক্ত। ভিজুয়্যাল বেসিক.নেটে ফাইল নিয়ে কাজ করার আরো একটি পদ্ধতি আছে। একে বলে **My.Computer.FileSystem Object** । নিম্নে এ সম্পর্কিত একটি উদাহরণ তুলে ধরছি। এসাথে শেষের দিকে একই কাজ সারতে প্রথম পদ্ধতি তথা **System.IO** -এর অন্তর্ভুক্ত কোডও দেওয়া হয়েছে। শিক্ষার্থীরা উভয় কোডের মধ্যে পার্থক্য লক্ষ্য করবেন।

এবার আমরা ধারাবাহিক পদক্ষেপে পুরো উদাহরণটি লিপিবদ্ধ করবো। শিক্ষার্থীরা নিচের প্রতিটি পদক্ষেপ সঠিকভাবে অনুসরণ করে এ প্রজেক্টটি তৈরি করুন।

ক. একটি নতুন উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। প্রজেক্টের নামকরণ করুন 'FolderAndFiles'। অবশ্য চাইলে অন্য নামও দিতে পারেন। প্রদর্শিত ফর্মের প্রোপার্টি হবে নিম্নরূপ:
খ. ফর্মে টুল বাক্স থেকে সংযুক্ত করুন একটি **ListBox,** একটি **CheckBox,** দু'টি **Button** এবং একটি **FolderBrowserDialog** কন্ট্রোল। টুল বাক্সে যেয়ে এগুলোর নামের ওপর ডবল ক্লিক করলেই ফর্মে এসে যুক্ত হবে। সকল কন্ট্রোলের নাম নিচের জানালার অনুরূপ করুন। আমি বাংলা নামকরণ করেছি। চাইলে আপনি অন্য নাম বা এমনকি ইংরেজিতেও নামকরণ করতে পারেন।



১১তম পরিচ্ছেদ: ***Working with Files***

ফাইল নিয়ে কাজ : *Working with Files*

ফর্মের বিভিন্ন বস্তুর ব্যাখ্যা: দেখতেই পাচ্ছেন, ফর্মে মোট ৫টি বস্তু আছে: ১. লিস্টবক্স [listBox1]- এই ফিল্ডে ফল্ডারে প্রাপ্ত ডকুমেন্টের প্যাথসহ নাম প্রদর্শিত হবে। ২.'ব্রাউজ' বোতাম [Button2] এর ক্লিক ইভেন্টে আমরা কোড লিখবো ফল্ডার চয়েজ করে তথ্য সংগ্রহ করতে এবং তা লিস্ট বাক্সে নিয়ে আসতে। ৩. 'ফাইল তথ্য'বোতাম [Button1] এর ক্লিক ইভেন্টে যে কোড দেওয়া হবে, তার দ্বারা আমরা লিস্ট বাক্স সিলেক্ট করা ফাইলের ওপর পুরো তথ্যাদি সংগ্রহ করে ম্যাসেজ বাক্সে দেখাবো। ৪. 'তথ্য সংরক্ষণ করুন' নামক চেক বাক্সের [checkBox1] মাধ্যমে আমরা ফাইল তথ্য বোতামে ক্লিক করার পর তা লগ [সংরক্ষণ] করবো কি না সে সিদ্ধান্ত নেবো। ব্যবহারকারী চেক বাক্সে ক্লিক করে থাকলে বুঝা যাবে সংরক্ষণ হবে, অন্যথায় নয়। ফর্মের বাইরে লাইনের নিচে 'FolderBrowserDialog1' নামক আরেকটি বস্তু আছে। এটার কাজ হলো কোডের মাধ্যমে ডাকা ফল্ডার ব্রাউজার

> ***নোট:*** কোড চলাকালে যে কোনো বস্তুকে ইঙ্গিত বা পারিচিত করা হয় তার Name Prperty-এর মাধ্যমে। যেমন: আমাদের এই উদাহরণে listBox1, Button1, Button2, checkBox1 এবং Form1। তবে বস্তুর টেক্সট প্রোপার্টি জানালায় প্রদর্শিত হবে। এটা যে কোনো কিছু হতে পারে- এতে কোডের ক্রিয়ার ওপর কোনো রদবদল আনবে না। সুতরাং আপনি চাইলে উভয় বোতাম, চেক বাক্স ও ফর্মের শিরোনামের টেক্সট প্রোপার্টি অন্য কিছু দিতে পারেন। আমরা এখানে বাংলা দিয়েছি- আপনি অন্য কোনো বাংলা টেক্সট বা এমনকি ইংরেজি টেক্সট্ও দিতে পারবেন - এতে কোডের একটি অক্ষরও বদলানো লাগবে না।

প্রদর্শন করা। এটা ফর্মের ভেতরে থাকার বস্তু নয়। যাক, ইউ.আই. ডিজাইনের কাজ শেষ হয়ে গেলো। এবার আমরা কোড লিখবো। আমরা পুরো কোডটিই এক সঙ্গে তুলে ধরছি। এরপর প্রত্যেক সাবরুটিনের ওপর ব্যাখ্যা দেবো।

```
Public Class Form1
    'where to start browsing? Set it to current directory
   Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      ' Set the default directory of the folder browser to the current directory.
      FolderBrowserDialog1.SelectedPath = My.Computer.FileSystem.CurrentDirectory
      SetEnabled() 'CheckBox1 and Button1 enabled
   End Sub
    'The FolderBrowserDialog1.ShowDialog call opens the Browse For Folder dialog box.
   'After the user clicks OK, the SelectedPath property is sent as an argument to the
   'ListFiles method [below].
   Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
      If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then
         ' List files in the folder.
         listFiles(FolderBrowserDialog1.SelectedPath)
      End If
      SetEnabled() 'CheckBox1 and Button1 enabled
   End Sub
   'The GetFiles method retrieves a collection of strings, one for each file in the directory.
   'The GetFiles method accepts a search pattern argument to retrieve files that match a particular pattern.
   'In this example, only files that have the extension .txt, .Doc and 'Docx [word files] are returned.
    'Returned string added to ListBox
   Private Sub ListFiles(ByVal folderPath As String)
      ListBox1.Items.Clear() 'clear any list in it
      Dim fileNames = My.Computer.FileSystem.GetFiles(folderPath, FileIO.SearchOption.SearchTopLevelOnly,
  "*.txt", "*.Doc", "*.Docx")
      For Each fileName As String In fileNames 'File names contained indexed file path
         ListBox1.Items.Add(fileName) 'build the list line by line
      Next
   End Sub
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If ListBox1.SelectedItem Is Nothing Then 'make sure user has selected an item
        MessageBox.Show("Please select a file.")
        Exit Sub
    End If
   Dim filePath = ListBox1.SelectedItem.ToString ' Obtain the file path from the list box selection.
    ' Verify that the file was not removed since the " button was clicked.
    If My.Computer.FileSystem.FileExists(filePath) = False Then
        MessageBox.Show("File Not Found: " & filePath)
        Exit Sub
    End If
    Dim fileInfoText As String = GetTextForOutput(filePath)  ' Obtain file information in a string.
    MessageBox.Show(fileInfoText) ' Show the file information.
    'The if-end if block of code sets the log file path to put it in the same directory as that of the selected file.
    'The text of the log entry is set to the current date and time followed by the file information.
    'The WriteAllText method, with the append argument set to True, is used to create the log entry.
    If CheckBox1.Checked = True Then
        ' Place the log file in the same folder as the examined file.
        Dim logFolder As String = My.Computer.FileSystem.GetFileInfo(filePath).DirectoryName
        Dim logFilePath = My.Computer.FileSystem.CombinePath(logFolder, "log.txt")
        Dim logText As String = "Logged: " & Date.Now.ToString & vbCrLf & fileInfoText & vbCrLf & vbCrLf
        ' Append text to the log file.
        My.Computer.FileSystem.WriteAllText(logFilePath, logText, append:=True)
    End If
End Sub
Private Function GetTextForOutput(ByVal filePath As String) As String
    If My.Computer.FileSystem.FileExists(filePath) = False Then
        Throw New Exception("File Not Found: " & filePath) ' Verify that the file exists. In no, show dialog & exit
    End If
    ' Create a new StringBuilder, which is used  to efficiently build strings.
    Dim sb As New System.Text.StringBuilder()
    ' Obtain file information.
    Dim thisFile As System.IO.FileInfo = My.Computer.FileSystem.GetFileInfo(filePath)
    ' Add file attributes.
    sb.Append("File: " & thisFile.FullName)
    sb.Append(vbCrLf)
    sb.Append("Modified: " & thisFile.LastWriteTime.ToString)
    sb.Append(vbCrLf)
    sb.Append("Size: " & thisFile.Length.ToString & " bytes")
    sb.Append(vbCrLf)
    ' Open the text file.
    Dim sr As System.IO.StreamReader =
        My.Computer.FileSystem.OpenTextFileReader(filePath)
    If sr.Peek() >= 0 Then
        sb.Append("First Line: " & sr.ReadLine()) ' Add the first line from the file.
    End If
    sr.Close()
    Return sb.ToString
End Function
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

```
'enable or disable Button1 and CheckBox wether ListBox1 item selected or not respectively
  Private Sub SetEnabled()
     'AnySelected becomes Yes or No depending wether any Item selected or not resectively
     Dim AnySelected As Boolean = (ListBox1.SelectedItem IsNot Nothing)
     Button1.Enabled = AnySelected 'if AnySelected True, Button1 is Enabled
     CheckBox1.Enabled = AnySelected 'if AnySelected True, CheckBox1 is Enabled too
  End Sub
  Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ListBox1.SelectedIndexChanged
     SetEnabled() 'CheckBox1 and Button1 enabled
  End Sub
End Class
```

কোডের ব্যাখ্যা: ফর্মে লোড ইভেন্ট হ্যান্ডলারে আমরা একটিমাত্র লাইন লিখেছি। লাইনটি এই:

FolderBrowserDialog1.SelectedPath = My.Computer.FileSystem.CurrentDirectory

এখানে ফল্ডারব্রাউজারডায়লগ ক্লাসের ডিফোল্ট প্যাথ বলে দেওয়া হয়েছে। ফাইল সিস্টেমের .CurrentDirectory যে ডাইরেক্টরী থেকে লিস্ট বাক্সে ফাইলের প্যাথ নিয়ে আসা হয়েছে সেই প্যাথের দিকেই নির্দেশ করে। অন্যকথায় প্রথমবার যে ডাইরেক্টরিতে যেয়ে আপনি ফল্ডার খুলবেন, সে-ই ডাইরেক্টরিকেই দ্বিতীয়বার দেখাবে। আর লগ টেক্সটটিও সেখানে সংরক্ষিত হবে। আমরা যদি My.Computer.FileSystem.CurrentDirectory এর বদলে এখানে **System.IO** ব্যবহার করি তাহলে লিখবো:

FolderBrowserDialog1.SelectedPath = System.IO.Directory.GetCurrentDirectory()

GetCurrentDirectory() -সিস্টেম.আইও এর একটি ফাংশন। এটা রিটার্ন করে বর্তমান ডাইরেক্টরির প্যাথ। সাবের দ্বিতীয় লাইনে অপর একটি সাবরুটিন কল করা হয়েছে। এ'টি 'ফাইল তথ্য' বোতাম ও 'তথ্য সংরক্ষণ করুন' চেক বাক্সকে অকার্যকর করে। এগুলো কার্যকর হবে এই একই সাব পুনরায় কল করে- যখন ব্যবহারকারী কোনো ফাইলের তথ্য জানার জন্য লিস্ট বাক্সের কোনো আইটেমে ক্লিক করবে।

Button2 -এর ক্লিক ইভেন্ট সাবে; If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then -অংশ প্রথমে ডায়লগটি দেখাবে। এরপর ব্যবহারকারী কোনো ফল্ডার চয়েজ করে 'OK' বোতামে ক্লিক করলে বুলিয়ান ফলাফল 'True' হবে। সুতরাং পরের লাইনে যেয়ে ListFile() সাবটি কল করবে- যার কাজ প্রাপ্ত ফা্ইলগুলোর নাম লিস্ট বাক্সে প্রদর্শন করা। আর যদি 'Cancel' বোতামে ক্লিক করা হয় তাহলে কিছুই হবে না। শুধুমাত্র ডায়লগটি বন্ধ হবে। একই সময় SetEnabled() সাব কল করে নিশ্চিত করা হবে 'ফাইল তথ্য' বোতাম ও তথ্য 'সংরক্ষণ করুন' চেক বাক্স যাতে Disabled অবস্থায় থাকে। অবশ্য ডায়লগের ফলাফল 'OK' কিংবা 'Cancel' হোক উভয় অবস্থায়ই Button2-তে ক্লিক করলে এ দু'টো Disabled থাকা জরুরী- আর তা-ই হবে। কারণ আমরা কল করছি **End If** কথাটির পরে।

ListFiles(ByVal folderPath As String) সাবের কাজ হলো প্রদত্ত ফল্ডার প্যাথ [সাবের আর্গুমেন্ট] ব্যবহার করে ঐ ফল্ডানের .txt, .Doc & .Docx ফাইল টাইপের যতো ফাইল আছে সবগুলোর প্যাথ সংগ্রহ করে লিস্ট বাক্সে প্রদর্শন করা। এজন্য অপারেটিং সিস্টেমের My.Computer.FileSystem.GetFiles ফাংশনটি কাজে লাগানো হয়েছে। এ ফাংশনের গুরুত্বপূর্ণ দু'টি আর্গুমেন্ট হলো: ফল্ডারের প্যাথ [যা আসবে FolderBrowserDialog1.SelectedPath থেকে] ও অনুসন্ধান ফিল্টার তথা [এখানে] .txt, .Doc & .Docx ফাইল টাইপ। অপর আর্গুমেন্ট নিশ্চিত করে শুধুমাত্র উচ্চ পর্যায়ের ডাইরেক্টরি থেকে তথ্য আনা হবে। অন্যকথায় অনুসন্ধান হবে শুধুমাত্র কারেন্ট ফল্ডারে- এর ভেতরস্থ অপর কোনো ফল্ডার এ অনুসন্ধানের আওতাভুক্ত থাকবে না।

১১তম পরিচ্ছেদ: ***Working with Files***

ফাইল নিয়ে কাজ : *Working with Files*

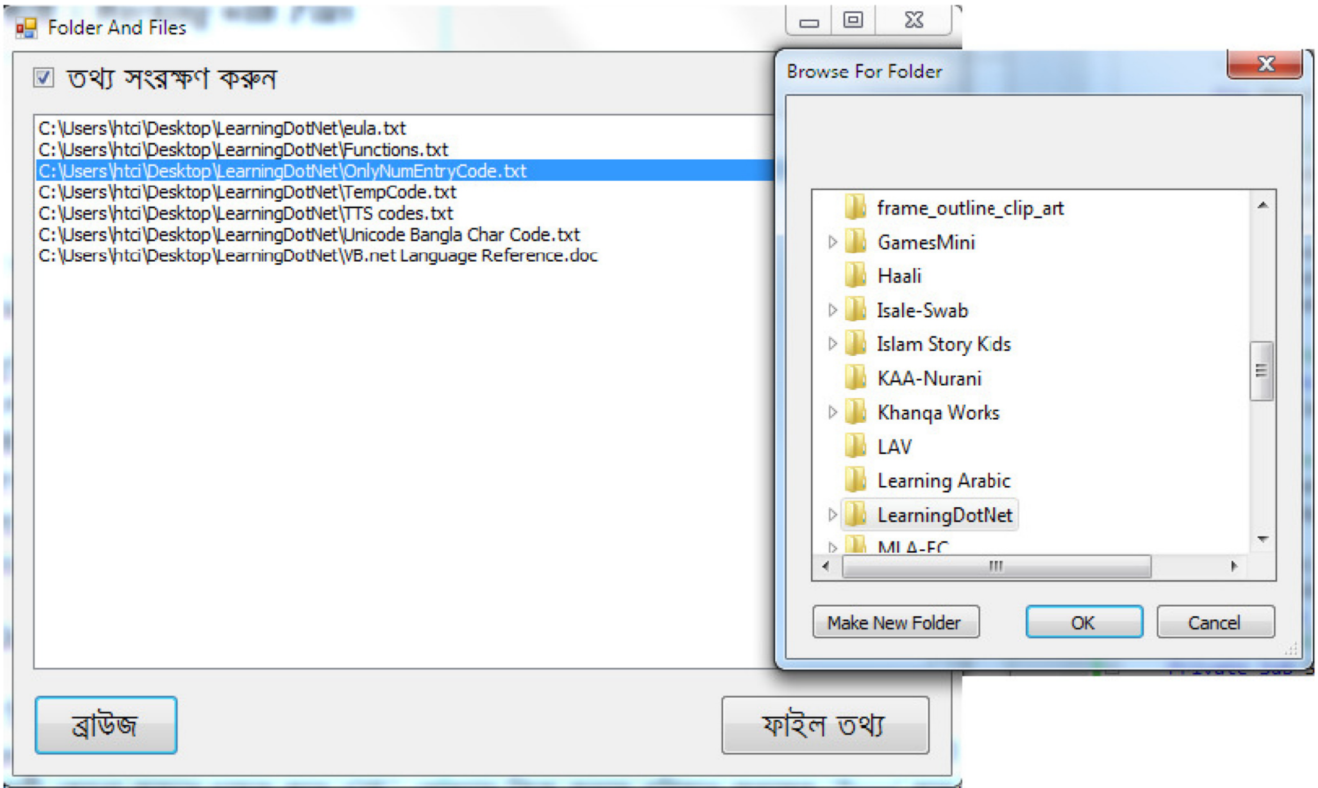
লিস্ট বাক্সে আইটেমগুলো লাইনের পর লাইনে লেখা হয় For Each - Next কোড অংশের মাধ্যমে।

Button1_Click ইভেন্ট সাবের কোডগুলো লিস্ট বাক্সের সিলেক্টেড ফাইল প্যাথে সংরক্ষিত ফাইলের তথ্যাদি একটি লগে [অর্থাৎ টেক্সট ফাইলে] সংরক্ষণ করবে কারেন্ট ডাইরেক্টরির ভেতর। প্রত্যেক বার কোনো ফাইলের প্যাথ সিলেক্ট করে এই বোতামে ক্লিক করলেই ঐ লগে তথ্যাদি যোগ হবে। সুতরাং লগ ফাইল একটি মাত্র তৈরি করা হবে।
সাবের প্রথম If-End If ব্লক নিশ্চিত করে লিস্ট বাক্সের কোনো আইটেম সিলেক্ট করা আছে কি না। না থাকলে মেসেজ জানালা এসে 'Please select a file.' কথাটি জানিয়ে কোডের কার্যক্রম এখানে শেষ হবে।
পরের লাইনে আমরা ফাইল প্যাথের লেখাটুকু একটি ভেরিয়েবলে সংরক্ষণ করেছি। এরপর আরেকটি If-End If ব্লকে নিশ্চিত করা হচ্ছে ফাইলটি আদৌ আছে কি না- কারণ, ইতোমধ্যে এটি মুছে যেতে পারে কিংবা অন্যত্র সরিয়ে নিতে পারেন ব্যবহারকারী। পরের লাইনে ফাইলের তথ্যাদি একটি ভেরিয়েবলে সংরক্ষণ করা হচ্ছে। এই তথ্যাদি আসবে আমাদের বানানো GetTextForOutput(filePath) নামক সাব থেকে। এ সাব সম্পর্কে একটু পরই ব্যাখ্যা তুলে ধরবো। পরের লাইনে মেসেজ বাক্সে প্রাপ্ত তথ্যাদি দেখানো হবে। এরপর আরেকটি If-End If ব্লকের ভেতর তথ্যাদি সংরক্ষণ করার জন্য লগ ফাইল বানানো হওয়ার কাজ সারা হচ্ছে। যদি চেক বাক্স ক্লিক করা থাকে তাহলেই এ কাজ হবে।
তথ্য সংরক্ষণের জন্য প্রথমে ডাইরেক্টরি তৈরি করে তা logFolder নামক ভেরিয়েবলে প্রাপ্ত টেক্সট রাখা হয়েছে। এ কাজের জন্য সিস্টেমের My.Computer.FileSystem.GetFileInfo(filePath).DirectoryName ক্লাসটি কল করেছি। এর পরের লাইনে লগ ফাইল সৃষ্টি হয়েছে। এর প্যাথটিও আমার logFilePath নামক ভেরিয়েবলে সংরক্ষণ করেছি। পরের লাইনে লগে কি লেখা হবে তা-ও নাকটি logText নামক ভেরিয়েবলে সংরক্ষণ করে লেখার কাজটি সেরেছি এই My.Computer.FileSystem.WriteAllText(logFilePath, logText, append:=True) কোড লাইনটি দ্বারা।
পরবর্তী তিনটি ব্লকের প্রথমটি ফাংশন এবং বাকি দু'টো সাবরুটিন। এগুলো প্রোগ্রামের অন্যত্র থেকে কল করা হয়।
ফাংশন GetTextForOutput কাজ হলো ফাইল প্যাথ [filePath] প্যারামিটার ব্যবহার করে ফা্ইল সম্পর্কে তথ্য সংগ্রহ করা ও তা টেক্সট হিসাবে কলিং কোডে ফিরে দেওয়া। এ কাজ আঞ্জাম দিতে ফাংশন ধারাবাহিকভাবে সিস্টেমের কয়েকটি গুরুত্বপূর্ণ মেথড কাজে লাগায়। যেমন:
১. My.Computer.FileSystem.FileExists(filePath) -এটা রিটার্ন করে একটি বুলিয়ান ভ্যালু। ফা্ইলের অস্তিত্ব থাকলে তা হবে True, অন্যথায় False।
2. New System.Text.StringBuilder() -এটা একটি সিস্টেম অবজেক্ট মেথড। এর কাজ হলো ফাইল সম্পর্কে বিভিন্ন তথ্যাদি যোগান দেওয়া: যেমন, ফাইলের পূর্ণাঙ্গ নাম, সর্বশেষ রদবদলের সময়, ফাইলের বাইট্স সংখ্যা [সাইজ] ইত্যাদি।
৩. My.Computer.FileSystem.GetFileInfo(filePath) -এই ফাংশন মেথড কল করে আমরা ফাইলের তথ্যাদির প্রতি পয়েন্ট করতে পারি।
৪. System.IO.StreamReader এবং My.Computer.FileSystem.OpenTextFileReader(filePath) ব্যবহার করে ফাইল খুলে পরের ব্লকে প্রথম লাইন পড়া এবং ফা্ইল বন্ধ করা।

এর পরের কোড ব্লক হলো লিস্ট বাক্সের SelectedIndexChanged ইভেন্ট হ্যান্ডলার ধরে SetEnabled সাবটি কল করা। আর এই সর্বশেষ সাবটির কাজ হলো 'তথ্য সংরক্ষণ করুন' চেক বাক্স ও 'ফাইল তথ্য' বোতামের Enabled প্রপার্টি সত্য হিসাবে নিশ্চিত করা। ব্যস্! এভাবেই কোডটি কাজ করে! পাঠকদের নিকট এই ব্যাখ্যা প্রথমত কঠিন মনে হবে। ভাববেন না, ধীরে ধীরে আপনি বুঝতে পারবেন ঠিকই। ইংরেজিতে একটি কথা আছে: Practice makes perfect। এ কথাটি ও নিজের বুঝার ক্ষমতার প্রতি আস্থা রাখুন। এ গ্রন্থের বিভিন্ন কোড উদাহরণ, প্রজেক্ট ইত্যাদি প্রয়োজনে বার বার স্টাডি করুন। আপনি অভিজ্ঞ প্রোগ্রামার বনতে মোটেই ব্যর্থ হবেন না। পরের পৃষ্ঠায় ফাইল নিয়ন্ত্রণের এই উদাহরণ প্রজেক্টটির রান-টাইম জানালা তুলে ধরেছি। একটু দেখে নিন।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*



উদাহরণ-১ -এর রানটাইম দু'টি জানালা।

ভিজুয়্যাল বেসিক.নেটে ফা্ইল নিয়ে কাজ করার উপর উপরোক্ত উদাহরণটি শিক্ষার্থীরা বার বার দেখে ও নিজে প্রজেক্টটি তৈরি করে ভালো করে অনুধাবনের চেষ্টা করবেন। আমরা এ্ই কোড দ্বারা দু'টি সিস্টেম ফাংশন ক্লাস সম্পর্কে অবগত হয়েছি। এগুলো হলো:

১. System.IO.File - যার ব্যবহার পরিচ্ছেদের প্রথম উদাহরণে দেখেছি।
২. My.Computer.FileSystem - যার ব্যবহার আমরা শেষের উদাহরণে দেখেছি।

উপরোক্ত উভয় ক্লাসের কাজ একই। আপনি যে কোনোটি বা উভয়টি একই এ্যাপলিকেশনে ব্যবহার করতে পারেন। তবে আমার ইচ্ছে হচ্ছে দ্বিতীয়টির ব্যবহারের উপর আলাদা একটি পরিচ্ছেদ রচনা করার। এর কার্যাকারিতা অনেক ব্যাপক। যেটুকু সম্ভব এ ক্লাস সম্বন্ধে জানা হবে বিশেষ উপকারী। তাই আসুন, পরের পরিচ্ছেদে এর ব্যবহারের আরো কিছু উদাহরণ দেখে নিই।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

My.Computer.Object

সিস্টেমের এই বস্তুটি আমাদেরকে সুযোগ দিয়েছে কম্পিউটারের বিভিন্ন প্রোপার্টিকে ব্যবহার ও নিয়ন্ত্রণ করার। যেমন: অডিও, ক্লিপবোড, ঘড়ি, ফাইল সিস্টেম, তথ্য, কিবোর্ড, মাউস, নেটওয়ার্ক, পোর্ট, রেজিস্ট্রি ইত্যাদি। এসব নিয়ন্ত্রণের কিছু উদাহরণ এবার উপস্থাপন করছি।

১. **My.Computer.Audio:** নিশ্চয়ই আপনি লক্ষ্য করেছেন, কোনো কোনো প্রোগ্রামের মধ্যে অডিও প্লে করতে। ভিজুয়্যাল বেসিক.নেটে এটা যে কোড দ্বারা করা হয় তা-ই হলো এই বস্তুটি। নিচের উদাহরণ দ্বারা ব্যবহার পদ্ধতি আমার শিখে নেবো।

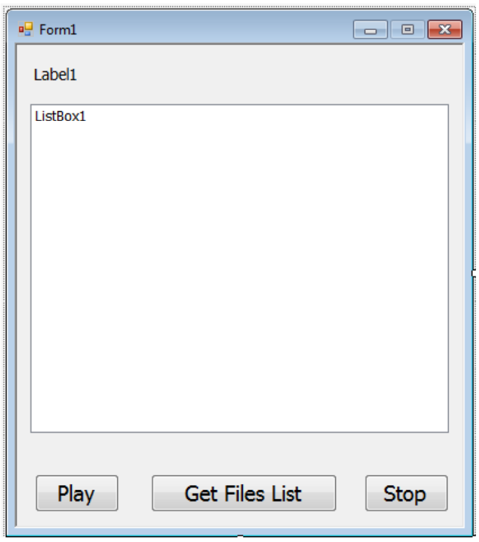
মনে করুন আমরা আল-কুরআনের একটি সূরা শুনবো। সূরাটির অডিও ফাইল সংরক্ষিত আছে যেখানে তার প্যাথ হলো: C:\Users\htci\Music\Muslima - Nas.mp3। আমরা এই ফাইলটি প্লে করতে পারি এই কোডটি দ্বারা:

```
Sub PlayBackgroundSoundFile()
   My.Computer.Audio.Play("C:\Users\htci\Music\Muslima - Nas.mp3",
AudioPlayMode.WaitToComplete) 'শেষের এই প্যারামিটার নিশ্চিত করে ফাইল প্লে শেষ না হওয়া পর্যন্ত
                        'পরবর্তী কোড রান করবে না।
End Sub
```

উপরোক্ত কোড ব্যবহার করে আমরা তৈরি করতে পারি একটি সাধারণ অডিও প্লেয়ার প্রোগ্রাম। আসুন তাহলে, এটা করে নিই।

উদাহরণ-১: Simple .wav audio player -.wav ফাইল অডিও প্লেয়ার

আপনার সকল অডিও ফাইলগুলো যে কোনো একটি ফল্ডারে সংরক্ষণ করে রাখতে পারেন। মনে রাখবেন ফাইলগুলো হতে হবে `.wav` ফাইল। আপনার অডিও ফাইল এই ফরমেটে না থাকলে ইন্টারনেট থেকে ফ্রিওয়্যার অডিও কনভার্টার ফাইল ডাউনলোড করতে পারেন। বেশ কয়েকটি ফ্রিওয়্যার আছে। আমি ব্যবহার করি- 'frea:ac - free audio converter v 1.0.16'। যাক, নিচের পদক্ষেপগুলো একে একে অনুসরণ করে প্রজেক্ট তৈরি করুন।



১. নতুন একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন- এর নামকরণ করুন: wavAudioPlayer।

২. ফর্মটির ডিজাইন পাশের ছবিতে প্রদর্শিত হয়েছে। এতে থাকবে একটি লেবেল, একটি লিস্ট বাক্স ও ৩টি বোতাম।

৩. Button1 ডবল ক্লিক করুন। ফর্মের ক্লাসের ভেতর বোতামের ক্লিক ইভেন্ট সাব আসবে। এতে লিখুন নিম্নের কোডটুকু।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
      If ListBox1.SelectedItem = "" Then Exit Sub 'make sure a file path is
chosen
      PlayFile() 'call sound play sub
End Sub
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

প্রথম লাইন দ্বারা নিশ্চিত করা হয়েছে যে, ব্যবহারকারী লিস্ট বাক্সের কোনো একটি লাইনে ক্লিক করে সিলেক্ট করেছেন। এটা নিশ্চিত হওয়ার পর পরের লাইনে আমরা নিম্নের সাবটি কল করেছি।

```
Sub PlayFile()
    'get the path string from list box selected line
    Dim FileP As String = ListBox1.SelectedItem
    My.Computer.Audio.Play(FileP) 'play file
    Label1.Text = "Now Playing: " & FileName(FileP) 'call FileName() sub show info
End Sub
```

উক্ত সাবের প্রথম লাইনে FileP নামক স্ট্রিং ভেরিয়েবলের মধ্যে লিস্ট বাক্সের ফাইল প্যাথ স্ট্রিংটি সংরক্ষণ করেছি। এরপর সিস্টেমের My.Computer.Audio.Play ফাংশনটি কল করেছি ও এর প্যারামিটার হিসাবে FileP -অর্থাৎ ফাইলের প্যাথ দিয়েছি। সুতরাং .wav ফাইলটি খুলে তা বাজতে শুরু হবে। পরের লাইনটি যে ফাইল প্লে হচ্ছে তার নাম লেবেলে দেখাবে। এবার লিখুন বোতাম২ এর ক্লিক ইভেন্ট সাবে নিচের কোড লাইনটি। এই কোড লাইন বলছে ফাইল প্লে করা বন্ধ করতে।

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    My.Computer.Audio.Stop() 'stops playing
End Sub
```

বোতাম৩ এর ক্লিক ইভেন্টে যে কোডটি দেওয়া হয়েছে তার দ্বারা উদ্দেশ্য হলো, ফল্ডার ব্রাউজার ডায়লগ খোলা ও যে কোনো ফল্ডারে প্রাপ্ত সকল .wav সাউন্ড ফাইলের প্যাথ স্ট্রিং এনে লিস্ট বাক্সে প্রদর্শন করা। সুতরাং নিচের কোড ব্লকটি বোতাম৩ এর ইভেন্ট হ্যান্ডলারে লিখতে হবে:

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
    If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then 'if Cancel, the code will not execute
        'List files in the folder.
        ListFiles(FolderBrowserDialog1.SelectedPath) 'call the sub below
    End If
End Sub
```

প্রথম লাইনের ‘=’ চিহ্নের আগের অংশ ফল্ডার ব্রাউজার ডায়লগটি খুলবে। এরপর ফলাফল যদি ‘OK’ হয়- অর্থাৎ ব্যবহারকারী কোনো ফল্ডার খুলেন তাহলে পরবর্তী লাইনের কোড রান করবে- অন্যথায় নয়। আমাদের তৈরি ListFiles() সাবরুটিনটি একটিমাত্র প্যারামিটার গ্রহণ করে- সেটা হলো সংরক্ষিত ফাইলের প্যাথ- যা আসবে ফল্ডার ব্রাউজার থেকে। এই সাবরুটিনটি নিম্নরূপ:

```
'opens all .wav files in a folder and list their path in the list Box
Private Sub ListFiles(ByVal folderPath As String)
      ListBox1.Items.Clear() 'clear any previous list
      Dim fileNames As String() = System.IO.Directory.GetFiles(folderPath, "*.wav", _
      System.IO.SearchOption.TopDirectoryOnly) 'this line gets the path list
      For Each fileName As String In fileNames 'this one puts the list line by line in the ListBox
         ListBox1.Items.Add(fileName)
      Next
End Sub
```

এই সাবরুটিনের প্রথম লাইন আগের কোনো লেখা থাকলে তা লিস্ট বাক্স থেকে মুছে ফেলে। দ্বিতীয় লাইন ফল্ডার ব্রাউজার থেকে সিলেক্ট করা ফল্ডারের ভেতর যতো .wav টাইপের সাউন্ড ফাইল আছে সবগুলোর প্যাথ সংগ্রহ করে। আর For Each -- Next লুপটির কাজ হলো সবগুলো ফাইলের প্যাথ স্ট্রিং, লাইনে লাইনে লিস্টবাক্সে লিখে দেওয়া। সর্বশেষ আমাদের বানানো ফাংশনটির ব্যবহার মূলত জরুরি নয়। তবে আমরা তো শিখবো! প্যাথ স্ট্রিং থেকে শুধু ফাইলের নাম কিভাবে বের করা যায় তা-ই দেখানো হলো এ ফাংশন তৈরির উদ্দেশ্যে।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

প্রথমে ফাংশনটি দেখে নিই। এরপর এর ব্যাখ্যা আসবে।

```
'this function returns the neame of a file from path string parameter
Private Function FileName(SupplyString As String) As String
    Dim A As Integer : Dim RetString As String = ""
    For A = Len(SupplyString) To 1 Step -1
        If Mid(SupplyString, A, 1) = "\" Then Exit For 'last '\' char position
    Next
    RetString = Mid(SupplyString, (A + 1), Len(SupplyString) - A) 'get the name
    Return RetString
End Function
```

ফাংশনের ব্যাখ্যা: প্রত্যেক প্যাথ নামের শেষের অংশে '\' -এ চিহ্নটি পরে ফাইলের নাম থাকে। সুতরাং আমরা চাই ফাংশনের প্যারামিটার হবে প্যাথ নামটি যার টাইপ হলো স্ট্রিং। ফাংশন থেকে যে ভ্যালু রিটার্ন হবে তা-ও একটি স্ট্রিং টাইপ। এজন্য আমরা প্রথম লাইনে লিখেছি:

Private Function FileName(SupplyString As String) As String

উক্ত লাইনের প্রথম দু'টি শব্দ লিখা বাধ্যতামূলক। এরপর এসেছে ফাংশনের নাম- এখানে FileName। বন্ধনীর ভেতর থাকবে প্যারামিটার- বা আর্গুমেন্ট যা কলিং কোডে দিতে হবে। কী দিতে হবে, তা আমরা সাব্যস্ত করে দিয়েছি- সেটা হলো স্ট্রিং টাইপ। এজন্য আমরা লিখেছি: (SupplyString As String)। এ তিনটি শব্দের প্রথমটি আমাদের বানানো, দ্বিতীয় ও তৃতীয়টি বাধ্যতামূলক- এখানে As String। কারণ, আমরা চাই SupplyString -এর টাইপ হবে স্ট্রিং। লাইনের শেষের দু'টি শব্দ দ্বারা আমরা জানিয়ে দিচ্ছি ফাংশন কোন্ টাইপের ভ্যালু রিটার্ন করবে- তা-হবে, স্ট্রিং। ফাংশনের দ্বিতীয় লাইনে দু'টি ভেরিয়েবল ঘোষণা করেছি: A - একটি ইন্টেজার টাইপ ও RetString -একটি স্ট্রিং টাইপ। শেষোক্তটি শূন্য স্ট্রিং হিসাবে ইনিশিয়েলাইজ করেছি। প্রথমটি পরবর্তী ফর-নেক্সট্ লুপটির জন্য ব্যবহৃত হয়েছে। এ ফর-নেক্সট্ লুপের প্রথম লাইন দ্বারা বুঝাচ্ছি, SupplyString তথা প্যাথ নামের দৈর্ঘ্যসংখ্যা থেকে ১ পর্যন্ত এই লুপটি রান করবে। আমাদের উদ্দেশ্য, ঠিক কোন্ পজিশনে স্ট্রিংয়ের মধ্যে শেষের '\' -এ চিহ্নটি আছে তা নির্ণয় করা। সুতরাং আমরা ডান থেকে শুরু করে একে একে বামদিকে যেয়ে উক্ত চিহ্নটির পজিশন খুঁজে দেখবো। যে মুহূর্তে এটা পাওয়া যাবে তখনই লুপ থেকে বের হবো। আর এ চিহ্নটি বার বার পরীক্ষার জন্য আমরা পরের লাইনের কোডটি ব্যবহার করেছি। তাহলো:

If Mid(SupplyString, A, 1) = "\" Then Exit For -এখানে, Mid ফাংশনের কাজ হলো স্ট্রিংয়ের পজিশন A -এর একটি মাত্র অক্ষর সংগ্রহ করা। আমরা বলছি এই অক্ষর বা ক্যারাক্টারটি যদি '\' হয়, তাহলে লুপ থেকে বের হও। মনে রাখবেন আমাদের ইন্টেজার ভেরিয়েবল A শুরু হয়েছে প্যাথ স্ট্রিং এর মোট অক্ষরসংখ্যা থেকে। আর এ সংখ্যার মান প্রত্যেক লুপে এক এক করে কমে আসছে। সুতরাং উক্ত '\' -ক্যারাক্টারটি যেখানে পাওয়া যাবে সেখানকার সংখ্যা-মান হবে A। অর্থাৎ ঠিক কোন্ সর্বডানের পজিশনে '\' -এ চিহ্নটি আছে তার মান আমরা পেলাম। এই মানটি বের করাই হলো ফাংশনের মূল কাজ।

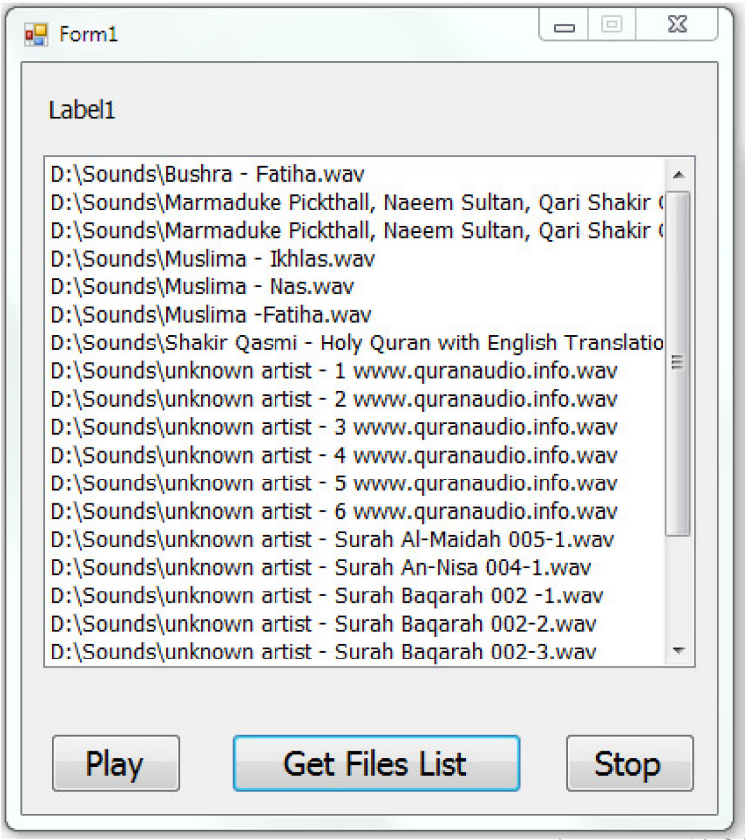
এবার আমরা ফাইলের নাম অংশ একটিমাত্র কোড লাইনে সংগ্রহ করতে পারি:

RetString = Mid(SupplyString, (A + 1), Len(SupplyString) - A) -এখানে Mid ফাংশনের কাজ হলো পুরো স্ট্রিং থেকে প্রাপ্ত সংখ্যা (A+1)-এ শুরু করে স্ট্রিং এর শেষ পর্যন্ত [Len(SupplyString) - A] সংগ্রহ করা। আমরা একে [অর্থাৎ ফাইলের নামকে] সংরক্ষণ করেছি RetString ভেরিয়েবলে। পরের লাইনে ফাংশন তার রিটার্ন ভ্যালু ফিরিয়ে দেবে। তাহলো এই সংরক্ষিত RetString ভেরিয়েবল: Return RetString।

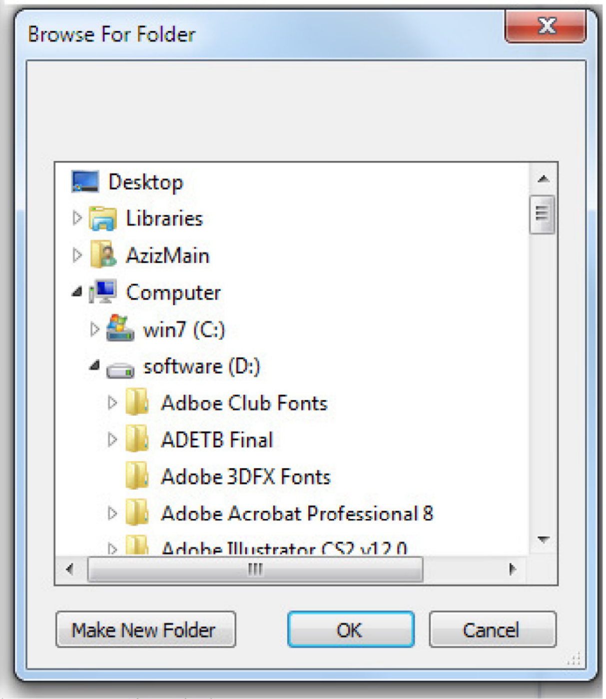
যাক, আশাকরি ফাংশনের ওপর এই বিস্তারিত ব্যাখ্যা আপনার জন্য ফাংশন কিভাবে কাজ করে তা বুঝতে সহায়ক হবে। পরের পৃষ্ঠায় একটি চিত্রে এ উদাহরণ প্রোগ্রামটির রান-টাইমের ছবি তুলে ধরেছি।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*



‘ফাইল অডিও প্লেয়ার’ উদাহরণের প্রোগ্রাম চলাকালের চিত্র।



উদাহরণ-২: Me.Computer -এর বিভিন্ন ফাংশন, মেথড ও প্রোপার্টি

Me.Computer-এর বিভিন্ন উদাহরণ একত্রে একটি প্রজেক্টে তুলে ধরেছি। পাঠকদের উচিৎ একটি উইন্ডোজ ফর্ম প্রজেক্ট সৃষ্টি করে নিচের পদক্ষেপ মুতাবিক এটি ধীরে ধীরে ডেভেলাপ করা।

(ক) My.Computer.Info.AvailablePhysicalMemory এটা একটি রিড-অনলি [শুধু পাঠ্য তথ্য] প্রোপার্টি। কম্পিউটারে কি পরিমাণ বাস্তব মুক্ত মেমোরি আছে তা এই প্রোপার্টির মাধ্যমে জানা যায়।

(খ) My.Computer.Info.TotalPhysicalMemory এটা মোট বাস্তব মেমোরি কত রিটার্ন করে।

(গ) My.Computer.Info.AvailableVirtualMemory এটা রিটার্ন করে মুক্ত ভারচুয়্যাল মেমোরির মাত্রা।

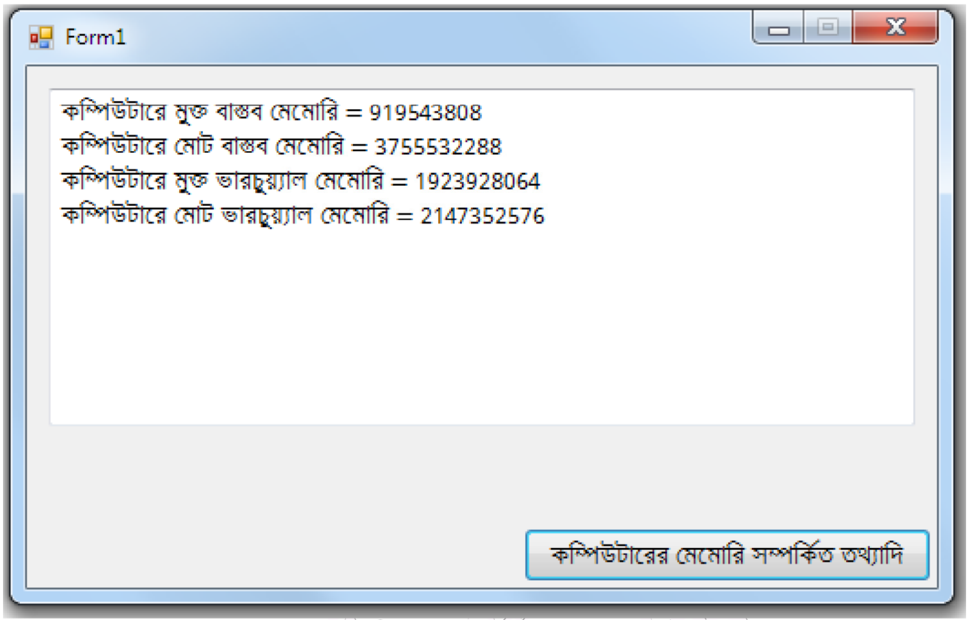
(ঘ) My.Computer.Info.TotalVirtualMemory এটা থেকে জানা যায় মোট ভারচুয়্যাল মেমোরি কত।

উপরোক্ত প্রোপার্টিগুলো কিভাবে কাজ করে দেখার জন্য প্রথমে প্রজেক্টের জানালায় একটি বোতাম ও একটি টেক্সট্ বাক্স যুক্ত করুন। এরপর বোতামের ক্লিক ইভেন্টে [বোতামে ডবল ক্লিক করলেই কোড জানালায় ক্লিক ইভেন্ট সাব আসবে] নিচের কোড ব্লকটি লিখুন।

```
    'information on the memories of user's computer in bytes.
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    TextBox1.Text = "" 'clear any text
    TextBox1.Text += ("কম্পিউটারে মুক্ত বাস্তব মেমোরি = " & _
    My.Computer.Info.AvailablePhysicalMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মোট বাস্তব মেমোরি = " & _
    My.Computer.Info.TotalPhysicalMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মুক্ত ভারচুয়্যাল মেমোরি = " & _
    My.Computer.Info.AvailableVirtualMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মোট ভারচুয়্যাল মেমোরি = " & _
    My.Computer.Info.TotalVirtualMemory)
End Sub
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*



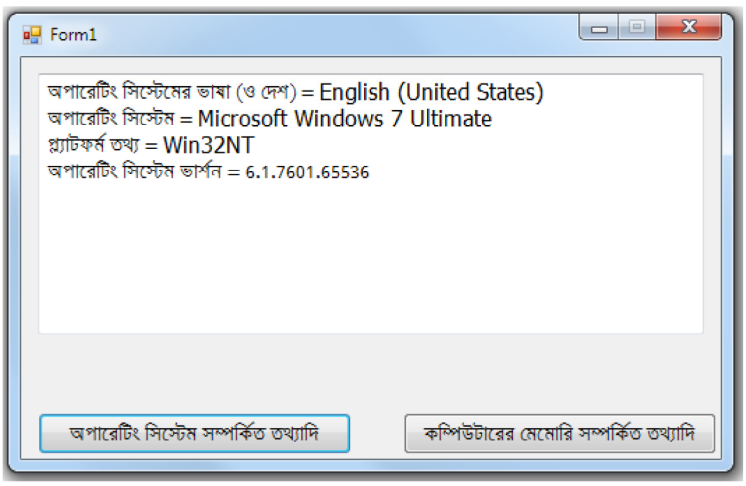
আগের পৃষ্ঠার কোড ব্লক ‘কম্পিউটারের মেমোরি সম্পর্কিত তথ্যাদি’ বোতামের ক্লিক ইভেন্টে ক্লিক করে রান করার ফলাফল।

(ঙ) My.Computer.Info.InstalledUICulture.DisplayName এটা দ্বারা কম্পিউটারের অপারেটিং সিস্টেমের ভাষা ও কোন্ দেশের ভাষা জানা যায়।
(চ) My.Computer.Info.OSFullName এটা অপারেটিং সিস্টেমের পূর্ণাঙ্গ নাম রিটার্ন করে।
(ছ) My.Computer.Info.OSPlatform এটা দ্বারা অপারেটিং সিস্টেমের প্ল্যাটফর্ম কি তা জানা যায়।
(জ) My.Computer.Info.OSVersion এটা দিয়ে আমরা জানতে পারি অপারেটিং সিস্টেমের ভার্শন কোন্‌টি।

উপরোক্ত প্রতিটি তথ্য জানতে ফর্মে আরেকটি বোতাম যুক্ত করুন। এর ক্লিক ইভেন্টে লিখুন:

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    TextBox1.Text = "" 'clear any text
    TextBox1.Text += ("অপারেটিং সিস্টেমের ভাষা (ও দেশ) = " & _
   My.Computer.Info.InstalledUICulture.DisplayName) & vbCrLf
    TextBox1.Text += ("অপরেটিং সিস্টেম = " & _
    My.Computer.Info.OSFullName) & vbCrLf
    TextBox1.Text += ("প্ল্যাটফর্ম তথ্য = " & _
    My.Computer.Info.OSPlatform) & vbCrLf
    TextBox1.Text += ("অপারেটিং সিস্টেম ভার্শন = " & _
    My.Computer.Info.OSVersion)
End Sub
```

লক্ষ্যণীয় যে, একই লাইনে দীর্ঘ কোড না লিখে আমরা (_) আন্ডারস্কোর দ্বারা লাইনটি ভাগ করে একাধিক লাইনে লিখতে পারি। উপরের কোডে চারটি লাইন এভাবে ভাগ করে জোড়া জোড়া লাইনে লিখেছি। আন্ডারস্কোর থাকতে হবে লাইনের শেষে। যাক, উপরের কোড রান করার ফলাফল ডানের চিত্রে তুলে ধরেছি। একটু দেখে নিন।



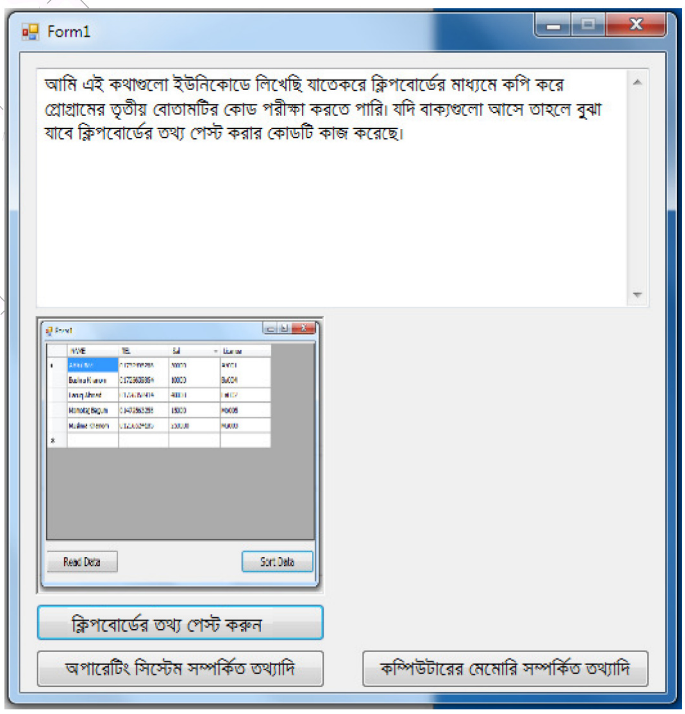
১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

(ঝ) My.Computer.Clipboard এই ফাংশনের মাধ্যমে আমরা ক্লিপবোর্ড সম্পর্কে তথ্য জানা ও একে নিয়ন্ত্রণ করতে পারি। উদাহরণ হিসাবে নিচের কোডগুলো দেখুন।

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    If My.Computer.Clipboard.ContainsText Then 'check if the clipboard contains text
        TextBox1.Text = ""
        TextBox1.Text = My.Computer.Clipboard.GetText 'paste into TextBox
    End If
    If My.Computer.Clipboard.ContainsImage Then 'check if the clipboard contains Image
        PictureBox1.Image = My.Computer.Clipboard.GetImage 'paste into PictureBox
    End If
End Sub
```

উক্ত সাবের ফাংশনগুলো কিভাবে কাজ করে তা দেখতে চাইলে আগের পৃষ্ঠায় প্রদর্শিত ফর্মে তৃতীয় একটি বোতাম যুক্ত করুন। আরো যুক্ত করুন একটি পিকচারবক্স। বোতামের টেক্সট যা ইচ্ছে দিতে পারেন পিকাচর বাক্সের SizeMode প্রোপার্টি হবে StretchImage । নতুন বোতামে ডবল ক্লিক করে উক্ত সাবটি লিখুন। এবার প্রোগ্রাম রান করুন। ক্লিপবোর্ডে ছবি বা লেখা কপি করুন যে কোনো ফাইল খুলে। ক্লিক করুন তৃতীয় বোতামে। ছবি থাকলে পিকচার বাক্সে আসবে, লেখা থাকলে টেক্সটবাক্সে আসবে। কিছু না থাকলে কিছুই আসবে না। নিচের ছবিতে প্রোগ্রাম রানকালে ক্লিপবোর্ড থেকে টেক্সট ও চিত্র আনার দৃশ্য দেখাচ্ছে।



(ঞ) My.Computer.Clock এই ক্লাসে ৩টি প্রপার্টি আছে যা আমরা কাজে লাগাতে পারি। এ প্রপার্টিগুলো হলো:

(১) My.Computer.Clock.GmtTime - উদাহরণ: MsgBox("Current GMT time: " & My.Computer.Clock.GmtTime) 'যে কম্পিউটারে এই কোডটি রান করবে, সেটির জিমএমটি [Greenwich Mean Time] সময় দেবে।

(২) **My.Computer.Clock.LocalTime** -এ প্রপার্টি থেকে স্থানীয় সময় পাওয়া যায়। যেমন:
MsgBox("Current local time: " & My.Computer.Clock.LocalTime)

(৩) **My.Computer.Clock.TickCount** -এ প্রপার্টি থেকে কম্পিউটারের ভেতরের টাইমারের মিলিসেকেন্ড গণনা পাওয়া যায়। যেমন:
Label1.Text = My.Computer.Clock.TickCount 'এ কোড রান করার সময় কম্পিউটারের অভ্যন্তরীণ টাইমারের মিলিসেকেন্ড সংখ্যা লেবেলে প্রদর্শিত হবে।

(ট) **My.Computer.Keyboard** -এই ক্লাসে বেশ ক'টি প্রপার্টি ও মেথড আছে যা আমরা কাজে লাগাতে পারি। কম্পিউটারের কিবোর্ড সম্পর্কিত কয়েকটি প্রপার্টি হলো:

(১) **My.Computer.Keyboard.AltKeyDown** -অলটার কি প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:
If My.Computer.Keyboard.AltKeyDown Then MsgBox("ALT key down")

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

(২) **My.Computer.Keyboard.CapsLock** -ক্যাপ্স লক প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:

```
If My.Computer.Keyboard.CapsLock Then MsgBox("Caps Lock is ON")
```

(৩) **My.Computer.Keyboard.CtrlKeyDown** -কন্ট্রোল কি প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:

```
If My.Computer.Keyboard.CtrlKeyDown Then MsgBox("Control key is down")
```

এভাবে আরো আছে, যাদের ফলাফল হচ্ছে বর্ণিত কি-টি চাপানো আছে কি না:

(৪) My.Computer.Keyboard.**NumLock**

(৫) My.Computer.Keyboard.**CtrlScrollLock**

(৬) এবং My.Computer.Keyboard.**ShiftKeyDown**

(ঠ) My.Computer.Mouse -এটির দ্বারা কম্পিউটারের মাউস সম্পর্কে জানা ও সেট করার দু'টি প্রোপার্টি জানা যায়। যেমন:

(১) **My.Computer.Mouse.WheelExists** -মাউসের মধ্যে স্ক্রল চাকা আছে কি না? থাকলে **'True'** না থাকলে **'False'**।

(২) **My.Computer.Mouse.WheelScrollLines** -মাউসের স্ক্রল চাকা প্রতি চক্করে কতটুকু ঘুরবে সে তথ্য দেবে। উদাহরণ হিসেবে নিচের কোড ব্লক ব্যবহার করে দেখুন। একটি বোতামের ইভেন্ট কোডে এটি লিখে রান করুন।

```
If My.Computer.Mouse.WheelExists Then
   Dim lines As Integer = My.Computer.Mouse.WheelScrollLines
   If lines > 0 Then
      MsgBox("Application scrolls " &
         lines & " line(s) for each wheel turn.")
   Else
      MsgBox("Application scrolls " &
         (-lines) & " page(s) for each wheel turn.")
   End If
Else
   MsgBox("Mouse has no scroll wheel.")
End If
```

(ঠ) My.Computer.Network - যে কম্পিউটারে কোড রান করবে তাতে নেটওয়ার্ক আছে কি না, থাকলে নির্দিষ্ট ইউআরএল ঠিকানা থেকে ফাইল ডাইনলোড করা ও ফাইল আপলোড করা ইত্যাদি কাজ আঞ্জাম দেওয়ার সুযোগ করে দিয়েছে এই মেথডটি। নিচে আমরা তিনটি ফাংশন উদাহরণসহ তুলে ধরেছি।

(১) My.Computer.Network.IsAvailable – এটা দ্বারা আমরা জানতে পারি বর্তমান সময়ে নেটওয়ার্ক আছে কি না? যেমন:

```
If My.Computer.Network.IsAvailable = True Then
   MsgBox("Computer is connected.")
Else
   MsgBox("Computer is not connected.")
End If
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

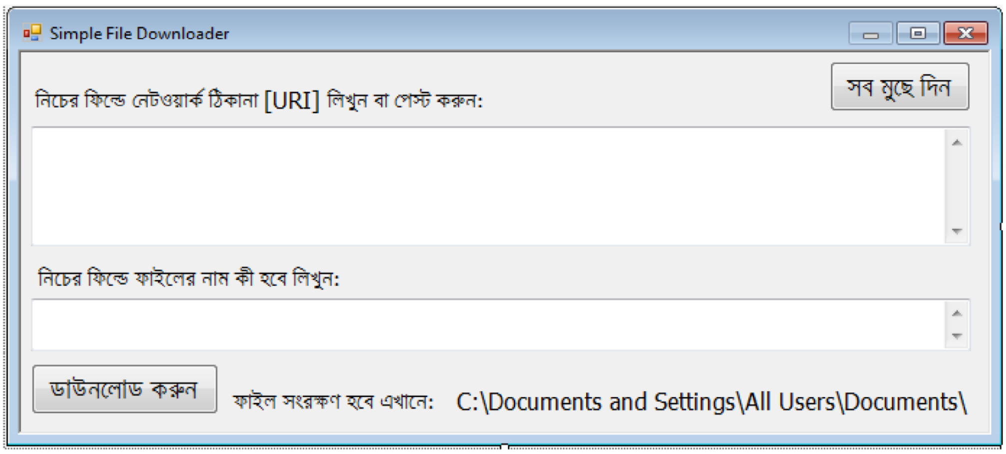
(২) **My.Computer.Network.DownlaodFile** -এই মেথডের কাজ হলো নির্দিষ্ট নেটওয়ার্ক ঠিকানার ফাইল ডাউনলোড করে নির্দিষ্ট জায়গায় সংরক্ষণ করা। যেমন:

```
My.Computer.Network.DownloadFile("http://sufiaekram.blogspot.com/p/junaid-baghdadira.html", "C:\Documents and Settings\All Users\Documents\Junaid Bagdadi (RA).html", "", "", True, 500, True)
```

উক্ত কোডের প্রথম আর্গুমেন্ট হলো ফাইলের ঠিকানা। দ্বিতীয়টি হলো কম্পিউটারের কোথায় তা ডাউনলোড হয়ে সংরক্ষিত হবে তার পূর্ণ প্যাথ। তৃতীয় ও চতুর্থ আর্গুমেন্ট হলো, যথাক্রমে ইউজার নেইম ও পাসওয়ার্ড [এ ক্ষেত্রে কিছুই নয় দেখানোর জন্য "" দেওয়া হয়েছে]। ৫ম আর্গুমেন্ট দ্বারা আমরা ডাউনলোড প্রগ্রেস বার দেখানোর নির্দেশ দিয়েছি। ৬ষ্ট আর্গুমেন্ট হলো টাইম-আউট লিমিট -এখানে ৫০০ মিলিসেকেন্ড এবং ৭ম আর্গুমেন্ট দ্বারা বলা হচ্ছে কম্পিউটারে এই একই ফাইল একই লকেশনে থাকলে তা অভাররাইট করা যাবে [এখানে False লিখলে অভাররাইট করা যাবে না- এই নির্দেশ দেওয়া হবে]।

আসুন, একটি ব্যবহারযোগ্য এ্যাপ্লিকেশন তৈরি করি, কেমন? আমাদের উদ্দেশ্য হলো, যে কোনো ওয়েভসাইট থেকে ডাউনলোড করা যায় এমন কোনো ফাইল ডাউনলোড করে আমাদের কম্পিউটারে সংরক্ষণ করতে। সুতরাং নিচের পদক্ষেপগুলো অনুসরণ করে আমরা প্রজেক্টটি তৈরি করবো।

ক. একটি নতুন প্রজেক্ট তৈরি করুন। এর নাকরণ হবে: SimpleFileDownloader।
খ. নিচের ছবি মুতাবিক ফর্ম ডিজাইন করুন। এতে থাকবে: ৪টি লেবেল, দু'টি টেক্সট্ বাক্স ও দু'টি বোতাম। ফর্মের FormBorderStyle হবে Fixed3D, টেক্সট্ প্রপার্টি হবে Simple File Downloader, এবং MaximizeBox প্রপার্টি হবে False।
গ. উপরের লেবেলের টেক্সট্ প্রপার্টিতে লিখুন: নিচের ফিল্ডে নেটওয়ার্ক ঠিকানা [URI] লিখুন বা পেস্ট করুন:। মাঝেরটিতে লিখুন: নিচের ফিল্ডে ফাইলের নাম কী হবে লিখুন:| নিচের বায়েরটিতে লিখুন: ফাইল সংরক্ষণ হবে এখানে:। সবশেষে নিচের ডানেরটিতে লিখুন: C:\Documents and Settings\All Users\Documents\।
নোট: আপনি চাইলে এসব লেবেলে ভিন্নভাবেও লিখতে পারেন- কিংবা ইংরেজিতেও লিখতে পারেন। আমি ইউনিকোডে বাংলায় লিখেছি। তবে শেষেরটিতে একটি প্যাথ নাম লিখেছি। ফাইল সংরক্ষণ করতে অন্য কোনো ফল্ডারও ব্যবহার করতে পারেন। সেক্ষেত্রে ফল্ডারের পূর্ণ প্যাথ নাম লিখতে হবে শেষোক্ত ফিল্ডে।
ঘ. উপরের ডানের বোতামের টেক্সট্ প্রপার্টিতে লিখুন: নতুন ফাইল ডাউনলোড করুন। দ্বিতীয়টির টেক্সট্ প্রপার্টিতে লিখুন: সব মুছে দিন| ব্যস! নিচের ছাবিতে দেখুন আমাদের ইউজার ইন্টারফেইস ডিজাইনটি।



১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

এবার বোতাম দু'টির ক্লিক ইভেন্টে আমরা কোড লিখবো। প্রথমে ডবল ক্লিক করুন 'সব মুছে দিন' বোতামে। ক্লিক ইভেন্টে নিচের কোডটি লিখুন।

```
'clear text for new entry
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
      TextBox1.Text = ""
      TextBox2.Text = ""
End Sub
```

উপরের কোডগুলো সহজেই অনুমেয়। আমরা উভয় টেক্সট বাক্স থেকে লেখা মুছে দিয়েছি। এবার ডবল ক্লিক করুন 'নতুন ফাইল ডাউনলোড করুন' বোতামে। ক্লিক ইভেন্টে লিখুন:

```
 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      Dim FIleURI As String = TextBox1.Text
      Dim FileSaveLoc As String = Label4.Text & TextBox2.Text
      If FIleURI = "" Or TextBox2.Text = "" Then
         MsgBox("Please Enter URI address or File Name. One or Both are MISSING!!",
MsgBoxStyle.Exclamation)
         Exit Sub
      End If
      On Error GoTo ThisLine
      My.Computer.Network.DownloadFile(FIleURI, FileSaveLoc, "", "", True, 500, True)
      MsgBox("File has been downloaded at: " & FileSaveLoc)
      Exit Sub
ThisLine:
      MsgBox("Sorry! An error occured. Please Try again. Make sure to enter correct texts
into fields.", MsgBoxStyle.Information)
   End Sub
```

দ্বিতীয় এ কোড ব্লকটিই মূল কোড। এগুলো লাইন বাই লাইন বুঝিয়ে বলছি। প্রথমত আমরা টেক্সটবাক্স-১ এ যে টেক্সটটুকু আছে তা একটি স্ট্রিং ভেরিয়েবলে সংরক্ষণ করেছি। অনুরূপ আরেকটি ভেরিয়েবলে সংরক্ষণ করেছি লেবেল৪ ও টেক্সট২ এর লেখাটুকু। লক্ষ করুন আমরা নির্দেশ দিচ্ছি লেবেলের টেক্সটের সাথে টেক্সটবক্স২ এর টেক্সট যোগ করতে। উভয়টি মিলে ফাইল সংরক্ষণের পূর্ণাঙ্গ প্যাথ সংরক্ষিত হবে FileSaveLoc ভেরিয়েবলে। ব্যবহারকারী কোনো একটি বা উভয় টেক্সট বাক্সে কিছু না লিখলে এ্যারর আসবে। তাই আমরা If - End If ব্লকে নিশ্চিত করছি ব্যবহারকারী উভয় ফিল্ডে টেক্সট লিখেছেন। না লিখলে একটি ম্যাসেজ আসবে। এরপর কোড রান করা এখানেই শেষ হবে।

On Error GoTo ThisLine -একটি পুরাতন কোডিং প্রাকটিস যা এখনো জনপ্রিয়। এখানে যা বলা হচ্ছে তাহলো: পরের কোড লাইনটিতে কোনো এ্যারর হলে ThisLine: -নামক কোডে জাম্প দাও। সেখায়ে যেয়ে একটি মেসেজ বাক্সে ভুলের রিপোর্ট প্রদর্শিত হবে। এরপর কোড রানিং শেষ হবে। এভাবে করার কারণ হলো, যাতে করে এ্যাররের ফলে পুরো এ্যাপলিকেশন বন্ধ বা ক্রাশ না করে।

যাক, যে কোড লাইনটি আসল কাজ তথা ফাইল ডাউনলোড করে তাহলো:

```
My.Computer.Network.DownloadFile(FIleURI, FileSaveLoc, "", "", True, 500, True)
```

আমরা ইতোমধ্যে জেনেছি এই মেথডের ৭টি আর্গুমেন্ট মানে কি। সুতরাং এখানে আর পুনরাবৃত্তি করছি না। তবে এটুকু বলছি যে, ব্যবহারকারী কর্তৃক লিখিত বা ওয়েবসাইট থেকে পেস্ট করা ফাইল হলো FIleURI। অনুরূপ ব্যবহারকারী কর্তৃক নির্দেশিত ফাইল নামসহ সংরক্ষণের পূর্ণ প্যাথ হলো FileSaveLoc।

১৩তম পরিচ্ছেদ: *Software Development Cycle*

সফ্‌টওয়্যার তৈরির প্রক্রিয়াচক্র : *Software Development Cycle*



ব্যবহারযোগ্য বা কমার্শিয়্যাল সফ্‌টওয়্যার তৈরি করতে প্রোগ্রামাররা একটি সুচিন্তিত প্রণালীবদ্ধ পদ্ধতি অনুসরণ করে থাকেন। এতে আছে বেশ ক'টি পদক্ষেপ। সঠিকভাবে এগুলো অনুসরণ করলে কাঙ্ক্ষিত ফলাফল পাওয়ার নিশ্চয়তা থাকে। আমরা এরূপ ৬টি পদক্ষেপ সনাক্ত করেছি যাকে, বলা যায়: সফ্‌টওয়্যার তৈরির প্রক্রিয়াচক্র (বায়ের চিত্রটি দেখুন)। ইংরেজিতে বলতে পারি:*Software Development Cycle*।

পরের দীর্ঘ পরিচ্ছেদে আমরা বেশ ক'টি ব্যবহারযোগ্য সফ্‌টওয়্যার ডেভেলাপ করবো। শিক্ষার্থীরা বায়ের চিত্রে প্রদর্শিত সব ক'টি পদক্ষেপ নিয়মতান্ত্রিণভাবে অনুসরণ করবেন- এটাই কাম্য। প্রত্যেক সফ্‌টওয়্যার তৈরির প্রক্রিয়া খুব স্পষ্ট ভাষায় ধারাবাহিকভাবে বর্ণিত হবে অবশ্যই। প্রথম প্রজেক্ট শুরুর আগে আসুন, আমরা এ ৬টি পদক্ষেপের ওপর আরো কিছু ব্যাখ্যা তুলে ধরি।

ক. প্রয়োজন সনাক্তকরণ - Identify Requirement

প্রত্যেক সফ্‌টওয়্যার সৃষ্টি করা হয় কোনো এক বা একাধিক প্রয়োজন মেটানোর উদ্দেশ্যে। যেমন: মাইক্রসফটের অফিস ২০০৭ সফ্‌টওয়্যারের মধ্যে মোট ৯টি ভিন্ন প্রোগ্রাম আছে। এগুলো হলো: Access, Excel, Groove, InfoPath, OneNote, Outlook, PowerPoint, Publisher and Word। এদের প্রত্যেকটির কাজ ভিন্ন। অফিস ডেভেলাপমেন্ট প্রোগ্রামাররা কম্পিটারের সাহায্যে বিভিন্ন কাজ কিভাবে আঞ্জাম দেওয়া যায় তা সনাক্ত করেছেন। কোন্ প্রয়োজন কোন্ প্রোগ্রাম মেটাবে তা সনাক্তকরণের নামই হলো Identify Requirement। আমাদের ১ম ব্যবহারযোগ্য প্রোগ্রামের নাম: 'বাংলাদেশ শহর থেকে শহর দূরত্ব' [BD City to City Distance]। এটি সৃষ্টির পূর্বে প্রয়োজন সনাক্তকরণ করতে যেয়ে আমরা দেশের ৬৪টি জিলা শহরের মধ্যে দূরত্ব কিলোমিটার ও মাইলের হিসাবে জানার স্পৃহার কথা ভেবেছি। সুতরাং এ প্রয়োজন মেটানোর জন্যই আমরা প্রোগ্রাম ডেভেলাপমেন্টে অগ্রসর হয়েছি।

খ. বিশ্লেষণ - Analysis

প্রয়োজন সনাক্তকরণের পর বুঝতে হবে কিভাবে তা মেটানো যায়। এজন্য প্রয়োজন সঠিক ব্যাখ্যার। আর এ পদক্ষেপের নামই হলো বিশ্লেষণ। আমাদের ১ম প্রোগ্রামের বিশ্লেষণ হলো: ব্যবহারকারী যে কোনো দু'টি শহরের নাম চয়েজ করে জেনে নেবেন এদের মধ্যে রোডের দূরত্ব কী। দূরত্বটি হবে কিলোমিটার ও মাইলের এককে।

গ. আকল্প - Design

উপরের উভয় পদক্ষেপগুলো থেকে সঠিক জবাব পাওয়ার পরই আসল ডেভেলাপমেন্ট প্রক্রিয়া শুরু হবে। প্রথমে অবশ্যই ইউজার ইন্টারফেইস তৈরি করতে হবে। আর এ ইউআই সৃষ্টির নামই হলো প্রোগ্রাম ডিজাইন পদক্ষেপ বা আকল্প। আমাদের ১ম প্রোগ্রামের ডিজাইনটি দেখলেই বুঝতে পারবেন এ স্টেজে আসলে কী করতে হয়। একটু পরই তা দেখতে পারবেন।

১৩তম পরিচ্ছেদ: *Software Development Cycle*

সফ্টওয়্যার তৈরির প্রক্রিয়াচক্র : *Software Development Cycle*

ঘ. সংক্ষেতলিপি - Coding

ডিজাইন স্টেজ শেষ করার পর প্রোগ্রামিংয়ের আসল কাজে হাত দিতে হবে, অর্থাৎ কোড লিখা। ব্যবহারকারী প্রোগ্রামের সাথে মিথস্ক্রিয়া করে কাঙ্ক্ষিত ফলাফল পাবে- এটাই তো সফ্টওয়্যার তৈরির মূল লক্ষ্য। অন্যকথায়, কাঙ্ক্ষিত ফলাফল দৃশ্যত হবে, কিংবা প্রিন্টারের মাধ্যমে কাগজে অঙ্কিত হবে বা ইন্টারনেটের মাধ্যমে অন্যত্র আপলোড করা হবে ইত্যাদি। তবে কাঙ্ক্ষিত ফলাফল অর্জনে প্রয়োজন হয় কিছু ক্রিয়া বা এ্যাকশনের। এসব ক্রিয়াকে সামগ্রিকভাবে বলা যায় **events** বা ঘটনাপ্রবাহ। বাস্তবে পুরো কম্পিউটার মূলত ইভেন্ট দ্বারা নিয়ন্ত্রিত হয়। প্রোগ্রামার হিসাবে আমরা এসব ইভেন্ট বা ঘটনাকে ক্যাচ্ করে নিজের ইচ্ছেমতো পরিচালনা করি। এটার নামই কোডিং। একটি উদাহরণ দেখুন। মনে করুন আপনার ফর্মে একটি বোতাম ও একটি লেবেল আছে। আপনার উদ্দেশ্য বোতামের ক্লিক নামক ইভেন্টকে কাজে লাগিয়ে ঐ লেবেলে কিছু লেখা প্রদর্শন করবেন। এ লক্ষ্যর্জনে ক্লিক ইভেন্টে এক লাইনের কোড লিখতে পারেন। যেমন:-

```
label1.text ="আমি বোতাম-১ এর ক্লিক ইভেন্ট ধরে এই কথাগুলো লিখেছি!"
```

প্রোগ্রাম চলাকালে বোতামে ক্লিক করলেই ইভেন্ট সক্রিয় হবে। আর যেহেতু আপনি একে ধরে বলছেন লেবেলে টেক্সট্ প্রদর্শনের জন্য, তাই এ কাজ করা ইভেন্টের জন্য বাধ্যতামূলক হয়ে যাবে।
অনুরূপ মাউস, কিবোর্ড, প্রিন্টার ইত্যাদি সবকিছু পরিচালনার জন্য একাধিক ইভেন্ট আছে। ভিজুয়্যাল বেসিক.নেট এসব ইভেন্ট ধরে কাজে লাগানের সুযোগ সৃষ্টি করে দিয়েছে। বিশেষ নিয়ম-পদ্ধতি অনুসরণ করে আপনি কোড লিখলে ইভেন্ট কাজে লাগিয়ে কাঙ্ক্ষিত ফলাফল পেতে পারেন।

ঙ. পরীক্ষণ - Testing

উপরে ব্যাখ্যাত কোডিং স্টেজটিই হলো সফ্টওয়্যার তৈরির মূল অংশ। এর পরের দু'টির প্রথমটি পরীক্ষণ স্টেজ। এ ব্যাপারে বেশি বলার প্রয়োজন নেই। কারণ, যে কোনো নতুন সৃষ্টি সঠিকভাবে ক্রিয়া করছে কি না তা বার বার পরীক্ষা না করে উপায় নেই। এমনকি কোডিং স্টেজেও বার বার প্রোগ্রাম রান (পরিচালনা) করে দেখতে হবে বিভিন্ন কোড ব্লক সঠিকভাবে কাজ করছে কি না? পুরো প্রোগ্রাম শেষ করার পরও বার বার টেস্ট করতে হয় বিভিন্ন অবস্থাভেদে। আপনি তো চাইবেন না, প্রোগ্রামের মধ্যে কেনো বাগ (খারাপ কোডিং) কিংবা এক্সেপেশন (ক্রাশ হওয়ার সম্ভাবনা) থাকুক! সুতরাং যে কোনো কন্ডিশনে যাতে আপনার প্রোগ্রামটি গ্রাহকের হাতে যেয়ে সঠিকভাবে চলে তা নিশ্চিত করতে যেয়ে আপনাকে বার বার প্রোগ্রাম পরিচালনা করে টেস্ট করতে হবে। এরূপ টেস্টিংকে বলে ডিবাগিং [debugging]।

চ. বিস্তারণ - Deployment

সর্বশেষ এ স্তরে পেঁছুনোর পরই বুঝতে হবে সফ্টওয়্যারটি সৃষ্টিতে সফলতা লাভ হয়েছে। কারণ, এখন একটি মাত্র কাজ বাকী রয়েছে। সেটা হলো, গ্রাহকের নিকট পৌঁছুনো। এ কাজের জন্য ভিজুয়্যাল স্টুডিও খুব সহজ একটি উপায় সৃষ্টি করে দিয়েছে। মাত্র কয়েকটি পদক্ষেপে আপনার সৃষ্ট সফ্টওয়্যারটি সিডি, পেন ড্রাইভ বা নেটওয়ার্কের মাধ্যমে বিস্তারণ করতে ইন্সটলার ফাইল বানিয়ে দেবে। আপনার সফ্টওয়্যারের জন্য প্রয়োজনীয় অন্যান্য ফাইলও প্যাকেজ করে দেবে যাতে করে ইন্সটলার ফাইল সহজে এগুলো গ্রাহকের কম্পিউটারে নির্দিষ্ট স্থানে সংরক্ষণ করে। এসব অতিরিক্ত ফাইলকে বলে **dependency files**। আপনার প্রোগ্রামে থাকতে পারে এক বা একাধিক টেক্সট্ ফাইল। প্রোগ্রাম চলাকালে এসব ফাইল থেকে তথ্য লোড করা জরুরী হতে পারে। সুতরাং এসব ফাইলও নির্দিষ্ট ফল্ডারে সংরক্ষিত থাকা প্রয়োজন।

যাক, এবার আমরা ১ম 'গ্রাহক ব্যবহারযোগ্য' একটি সফ্টওয়্যার তৈরির কাজে অগ্রসর হবো। শিক্ষার্থীদের ধৈর্যধারণ করা একান্ত কাম্য- কারণ, ১ম এই সফ্টওয়্যারের জন্য কয়েক শত লাইন কোডিং করতে হবে! তবে হতাশ হবেন না, অধিকাংশ ক্ষেত্রে কোড হবে কপি-পেস্ট! অবশ্য বেশ কয়েক পৃষ্ঠাব্যাপী প্রোগ্রামের ব্যাখ্য এসেছে- ধৈর্য ও মনোযোগসহ এ পৃষ্ঠাগুলো পাঠ করতে হবে। সেসাথে আইডিই খুলে নির্দেশিত পদক্ষেপ অনুসরণ করে সফ্টওয়্যার বানিয়েও যেতে হবে।

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

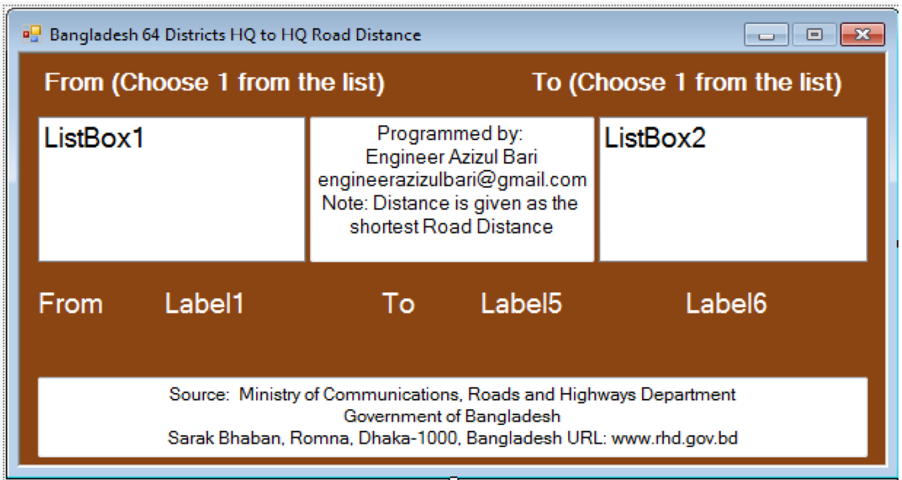
Application 1: BD City to City Distance - বাংলাদেশ শহর থেকে শহর দূরত্ব

উদ্দেশ্য: আমাদের উদ্দেশ্য হলো, ব্যবহারকারী বাংলাদেশের ৬৪টি জিলা শহরের রোড-দূরত্ব সহজে জানতে পারবেন। দূরত্বে প্রদর্শিত হবে কিলোমিটার ও মাইল এককে।

ডিজাইন: প্রথমেই আমরা একটি আকর্ষণীয় ইউজার ইন্টারফেইস ডিজাইন করবো। বায়ের চিত্রটি দেখুন ও পদক্ষেপগুলো অনুসরণ করুন।

১. একটি নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন: “BD City to City Distance”।

২. বায়ের চিত্রের মতো ফর্মটি ডিজাইন করুন।



ফর্মের বস্তুসমূহ: ৭টি লেবেল, ২টি লিস্ট-বাক্স এবং (চাইলে) দু’টি টেক্স-বাক্স।

ফর্মটি যাতে একই মাপের থাকে সেজন্য এর FormBorderStyle = Fixed3D, MaximizeBox = False and AutoScaleMode = None হবে। এর Text প্রপার্টি হবে Bangladesh 64 Districts HQ to HQ Road Distance। আপনি চাইলে অন্য নামও দিতে পারেন। রং আমি একটা চয়েজ করেছি। আপনি যা চান সেটাই দিতে পারেন। সবশেষে দু’টি ফিল্ডে কিছু তথ্য দিয়েছি। নিচেরটি জরুরী বলে মনে করি। আর উপরেরটি না দিলেই হবে। অবশ্য থাকলে তো আপনার নিজের নামই থাকবে, আমার পরিবর্তে! বিভিন্ন লেবেলে যা লিখা আছে তা-ই (টেক্স্ট) প্রোপার্টিতে লিখুন। ব্যস! ডিজাইন হয়ে গেলো। এবার আসল কাজ- কোডিং!

প্রোগ্রামের তথ্যাদি: আমাদের এই বিশেষ প্রোগ্রামটি কিছু তথ্যের উপর নির্ভরশীল। আগে থেকেই আমাদেরকে ৬৪টি জিলার নামসহ একটা থেকে আরেকটির দূরত্ব (কিলোমিটার এককে) জানতে হবে। এজন্য আমরা যোগাযোগ মন্ত্রণালয়ের ওয়েভসাইটটি কাজে লাগিয়েছি। সেখানে দেওয়া আছে একটি টেবিল। এতে জিলাগুলোর নাম ও একটা থেকে আরেকটির দূরত্ব মেট্রিক্স পদ্ধতিতে লিখিত আছে। এই তথ্যগুলোর উপর নির্ভর করেই আমরা পুরো এ্যাপলিকেশনটি তৈরি করেছি। একজন ব্যবহারকারী প্রশ্ন করতে পারেন: দূরত্ব আপনি পেলেন কোথায়? এগুলো কি সঠিক? আমাদেরকে জবাব দিতে হবে না। বাংলাদেশের তথ্য মন্ত্রণালয় থেকেই যে তথ্যগুলো পেয়েছি তা-তো আমরা নিচের টেক্স বাক্সে ঘোষণা দিয়েছি! যাক এবার আসুন কোডগুলোর ওপর আলোচনা শুরু করি। আপনি শহরের নাম ও দূরত্ব সংখ্যা টাইপ করতে সতর্কতা অবলম্বন করবেন- তথ্যে ভুল হলে হয় প্রোগ্রাম কাজ করবে না, কিংবা ভুল ফলাফল পাবেন। আমরা এর কোনোটাই চাই না।

২. কোডিং: ফর্মে ডবল-ক্লিক করে কোড জানালা খুলুন। প্রথমেই আমাদেরকে কয়েকটি গ্লোব্যাল ভেরিয়েবল ডিক্লায়ার করতে হবে। নিচের (পরের পৃষ্ঠায় দেওয়া) কোড লাইন মুতাবিক এগুলো ডিক্লায়ার করুন।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Public Class Form1
    'put distance of citys in array variables as matrices - i.e. 0 km to 0 index as distance to other city: Dhaka 0 to Bhola 205 (both 0 index)
    Dim Dhaka() As String = {"0", "178", "316", "247", "169", "205", "197", "109", "115", "242", "215", "96", "391", "338", "101", "149", "268", "37", "127", "163", "179", "164", "182", "178", "249", "259", "180", "117", "348", "183", "137", "343", "90", "150", "63", "240", "203", "27", "122", "247", "130", "17", "51", "210", "302", "158", "359", "158", "216", "443", "204", "185", "118", "256", "293", "304", "240", "101", "188", "134", "296", "241", "92", "407"}
```

লক্ষ করুন: Dhaka() একটি স্ট্রিং এ্যারে ভেরিয়েবল যার মধ্যে আছে ৬৪টি ইলেমেন্ট (ইনডেক্স ০ থেকে ৬৩)। প্রোগ্রাম চলাকালে এই ভেরিয়েবল [এবং শহরগুলোর নামে যাবতীয় এ্যারে ভেরিয়েবল] মেমোরিতে সংরক্ষিত থাকবে এবং এগুলো প্রোগ্রামের যে কোনো জায়গা থেকে ব্যবহারও করা যাবে- যা আমাদের ইচ্ছে। এবার বাকী শহরগুলোর এ্যারে ভেরিবেল ডিক্লায়ার করুন:

```
    Dim Bagerhat() As String = {"178", "0", "437", "160", "119", "155", "328", "283", "182", "363", "175", "269", "512", "467", "140", "270", "397", "215", "52", "336", "357", "93", "100", "139", "378", "381", "34", "290", "477", "184", "209", "472", "110", "136", "200", "201", "376", "179", "300", "357", "107", "186", "225", "260", "352", "336", "488", "244", "234", "572", "151", "66", "167", "306", "415", "433", "94", "134", "366", "311", "470", "414", "270", "536"}
    Dim Bandarban() As String = {"316", "437", "0", "469", "391", "427", "510", "301", "271", "85", "526", "225", "121", "651", "413", "167", "581", "350", "386", "376", "492", "476", "404", "490", "563", "133", "439", "379", "661", "499", "229", "656", "327", "461", "379", "552", "416", "329", "436", "561", "442", "314", "339", "524", "616", "444", "672", "208", "530", "757", "426", "438", "434", "569", "74", "618", "499", "317", "501", "447", "509", "454", "406", "720"}
    Dim Barguna() As String = {"247", "160", "469", "0", "84", "120", "445", "352", "214", "396", "323", "281", "544", "584", "209", "303", "513", "284", "196", "406", "426", "209", "95", "286", "495", "413", "150", "360", "594", "301", "241", "589", "143", "258", "270", "348", "446", "248", "370", "473", "227", "255", "294", "377", "469", "406", "604", "276", "350", "689", "44", "96", "236", "423", "447", "550", "206", "166", "435", "381", "539", "483", "340", "652"}
    Dim Barisal() As String = {"169", "119", "391", "84", "0", "36", "366", "273", "135", "317", "244", "202", "465", "505", "131", "224", "435", "206", "117", "327", "348", "168", "19", "207", "416", "334", "109", "281", "515", "222", "162", "510", "64", "179", "191", "270", "367", "169", "291", "395", "148", "177", "215", "298", "390", "327", "526", "197", "272", "610", "41", "53", "158", "344", "368", "471", "164", "87", "357", "302", "460", "405", "261", "574"}
    Dim Bhola() As String = {"205", "155", "427", "120", "36", "0", "402", "309", "171", "353", "280", "238", "501", "541", "167", "260", "471", "242", "153", "363", "384", "204", "55", "243", "452", "370", "145", "317", "551", "258", "198", "546", "100", "215", "227", "306", "403", "205", "327", "431", "184", "213", "251", "334", "426", "363", "562", "233", "308", "646", "77", "89", "194", "380", "404", "507", "200", "123", "393", "338", "496", "441", "297", "610"}
    Dim Bogra() As String = {"197", "328", "510", "445", "366", "402", "0", "274", "309", "437", "192", "291", "585", "141", "239", "344", "71", "169", "324", "328", "172", "235", "379", "190", "52", "454", "295", "201", "151", "144", "331", "146", "287", "217", "188", "182", "368", "221", "178", "50", "269", "211", "217", "68", "160", "217", "162", "353", "127", "246", "401", "350", "208", "114", "488", "107", "307", "298", "188", "72", "462", "406", "105", "210"}
    Dim Brahmanbaria() As String = {"109", "283", "301", "352", "273", "309", "274", "0", "143", "227", "320", "81", "375", "416", "206", "134", "345", "114", "231", "76", "203", "269", "287", "283", "327", "244", "285", "79", "425", "268", "164", "420", "195", "254", "148", "326", "116", "123", "147", "325", "235", "107", "67", "288", "380", "144", "436", "143", "294", "521", "308", "290", "204", "334", "278", "382", "344", "205", "204", "211", "209", "153", "170", "484"}
    Dim Chandpur() As String = {"115", "182", "271", "214", "135", "171", "309", "143", "0", "198", "257", "67", "346", "450", "144", "105", "380", "149", "130", "218", "291", "195", "149", "220", "361", "215", "184", "221", "460", "235", "43", "455", "71", "192", "177", "283", "258", "128", "234", "359", "162", "113", "137", "311", "403", "270", "471", "78", "285", "555", "170", "182", "171", "357", "249", "416", "243", "62", "300", "246", "351", "295", "204", "519"}
 Dim Chittagong() As String = {"242", "363", "85", "396", "317", "353", "437", "227", "198", "0", "453", "151", "160", "578", "339", "93", "507", "276", "312", "302", "418", "402", "330", "416", "489", "117", "366", "305", "588", "425", "155", "583", "253", "387", "305", "478", "342", "256", "362", "487", "368", "240", "265", "450", "542", "370", "598", "134", "456", "683", "352", "364", "360", "496", "78", "544", "425", "244", "427", "373", "436", "380", "332", "646"}
    Dim Chuadanga() As String = {"215", "175", "526", "323", "244", "280", "192", "320", "257", "453", "0", "306", "601", "331", "118", "360", "261", "237", "171", "373", "276", "82", "231", "37", "243", "470", "142", "305", "341", "49", "284", "336", "186", "65", "173", "26", "413", "216", "282", "221", "116", "223", "262", "124", "217", "321", "352", "319", "98", "437", "279", "198", "108", "170", "504", "297", "154", "209", "291", "176", "507", "451", "209", "400"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Comilla() As String = {"96", "269", "225", "281", "202", "238", "291", "81", "67", "151", "306", "0", "300", "432", "193", "58", "361", "130", "218", "156", "272", "256", "215", "270", "343", "169", "272", "159", "441", "279", "89", "436", "138", "241", "159", "332", "196", "109", "216", "341", "222", "94", "119", "304", "396", "224", "452", "67", "310", "537", "237", "249", "214", "350", "203", "398", "331", "128", "281", "227", "290", "234", "186", "500"}
Dim coxsBazar() As String = {"391", "512", "121", "544", "465", "501", "585", "375", "346", "160", "601", "300", "0", "726", "487", "241", "655", "425", "460", "450", "567", "550", "479", "564", "637", "252", "514", "453", "736", "573", "303", "731", "401", "536", "453", "626", "490", "404", "510", "635", "516", "389", "413", "598", "690", "518", "747", "282", "604", "831", "500", "512", "509", "644", "195", "692", "574", "392", "576", "522", "584", "528", "480", "795"}
Dim Dinajpur() As String = {"338", "467", "651", "584", "505", "541", "141", "416", "450", "578", "331", "432", "726", "0", "378", "485", "120", "310", "463", "469", "313", "374", "519", "329", "89", "595", "434", "342", "134", "283", "472", "129", "428", "355", "329", "321", "509", "362", "320", "127", "408", "352", "358", "207", "216", "359", "57", "494", "266", "94", "540", "489", "347", "205", "629", "78", "446", "439", "329", "213", "603", "547", "246", "58"}
Dim Faridpur() As String = {"101", "140", "413", "209", "131", "167", "239", "206", "144", "339", "118", "193", "487", "378", "0", "246", "308", "128", "89", "260", "235", "96", "144", "81", "290", "356", "155", "214", "388", "96", "171", "383", "73", "52", "65", "143", "300", "102", "203", "268", "100", "109", "148", "171", "264", "239", "399", "206", "145", "484", "166", "147", "31", "217", "390", "345", "168", "96", "250", "190", "393", "337", "148", "447"}
Dim Feni() As String = {"149", "270", "167", "303", "224", "260", "344", "134", "105", "93", "360", "58", "241", "485", "246", "0", "414", "183", "219", "209", "325", "309", "237", "323", "396", "110", "273", "212", "495", "332", "62", "490", "160", "294", "212", "385", "249", "163", "269", "394", "275", "147", "172", "357", "449", "277", "505", "41", "363", "590", "259", "271", "267", "403", "144", "451", "332", "151", "334", "280", "343", "287", "239", "553"}
Dim Gaibandha() As String = {"268", "397", "581", "513", "435", "471", "71", "345", "380", "507", "261", "361", "655", "120", "308", "414", "0", "239", "392", "399", "243", "304", "448", "259", "77", "524", "363", "271", "120", "212", "402", "115", "358", "285", "259", "250", "439", "292", "249", "117", "338", "282", "287", "137", "229", "288", "131", "423", "195", "216", "470", "419", "277", "182", "558", "76", "376", "368", "258", "142", "532", "476", "176", "179"}
Dim Gazipur() As String = {"37", "215", "350", "284", "206", "242", "169", "114", "149", "276", "237", "130", "425", "310", "128", "183", "239", "0", "164", "168", "151", "216", "219", "200", "221", "294", "217", "99", "320", "184", "171", "315", "127", "172", "64", "241", "208", "61", "93", "219", "167", "51", "56", "182", "274", "129", "330", "193", "188", "415", "241", "222", "119", "228", "328", "276", "287", "138", "159", "105", "301", "246", "64", "378"}
Dim Gopalganj() As String = {"127", "52", "386", "196", "117", "153", "324", "231", "130", "312", "171", "218", "460", "463", "89", "219", "392", "164", "0", "285", "305", "89", "91", "135", "374", "329", "54", "239", "473", "180", "157", "468", "59", "105", "149", "197", "325", "127", "249", "353", "55", "134", "173", "256", "348", "285", "483", "192", "230", "568", "152", "59", "116", "302", "363", "429", "114", "82", "314", "260", "418", "363", "219", "532"}
Dim Habiganj() As String = {"163", "336", "376", "406", "327", "363", "328", "76", "218", "302", "373", "156", "450", "469", "260", "209", "399", "168", "285", "0", "257", "322", "340", "336", "380", "319", "338", "132", "479", "322", "239", "474", "248", "308", "202", "380", "63", "176", "200", "378", "289", "161", "121", "341", "433", "197", "490", "218", "347", "574", "362", "343", "257", "387", "353", "435", "398", "259", "258", "265", "137", "81", "223", "538"}
Dim Jamalpur() As String = {"179", "357", "492", "426", "348", "384", "172", "203", "291", "418", "276", "272", "567", "313", "235", "325", "243", "151", "305", "257", "0", "319", "361", "274", "224", "436", "378", "125", "323", "227", "313", "318", "269", "278", "170", "266", "297", "203", "57", "222", "309", "193", "198", "185", "278", "96", "334", "335", "191", "418", "383", "364", "226", "231", "470", "279", "391", "280", "16", "109", "391", "335", "87", "382"}
Dim Jessore() As String = {"164", "93", "476", "209", "168", "204", "235", "269", "195", "402", "82", "256", "550", "374", "96", "309", "304", "216", "89", "322", "319", "0", "148", "46", "286", "419", "59", "277", "384", "92", "222", "379", "124", "44", "152", "108", "362", "165", "291", "264", "34", "172", "211", "167", "259", "327", "395", "257", "141", "480", "200", "115", "119", "213", "453", "340", "72", "147", "334", "219", "456", "400", "236", "443"}
Dim Jhalokathi() As String = {"182", "100", "404", "95", "19", "55", "379", "287", "149", "330", "231", "215", "479", "519", "144", "237", "448", "219", "91", "340", "361", "148", "0", "194", "430", "348", "90", "294", "528", "236", "176", "523", "77", "193", "205", "256", "380", "183", "305", "408", "162", "190", "229", "311", "404", "340", "539", "210", "285", "624", "52", "33", "171", "357", "382", "485", "145", "101", "370", "316", "474", "418", "275", "587"}
Dim Jhenaidah() As String = {"178", "139", "490", "286", "207", "243", "190", "283", "220", "416", "37", "270", "564", "329", "81", "323", "259", "200", "135", "336", "274", "46", "194", "0", "240", "433", "105", "281", "339", "46", "248", "334", "149", "28", "137", "62", "376", "179", "280", "219", "79", "186", "225", "122", "214", "319", "350", "282", "96", "434", "242", "161", "103", "168", "467", "295", "118", "173", "289", "173", "470", "414", "207", "398"}
Dim Joypurhat() As String = {"249", "378", "563", "495", "416", "452", "52", "327", "361", "489", "243", "343", "637", "89", "290", "396", "77", "221", "374", "380", "224", "286", "430", "240", "0", "506", "345", "253", "157", "194", "384", "152", "340", "267", "241", "232", "420", "274", "231", "39", "319", "264", "269", "118", "128", "270", "103", "405", "177", "181", "451", "401", "259", "116", "540", "113", "358", "350", "240", "124", "514", "458", "158", "144"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Khagrachhari() As String = {"259", "381", "133", "413", "334", "370", "454", "244", "215", "117", "470", "169", "252", "595", "356", "110", "524", "294", "329", "319", "436", "419", "348", "433", "506", "0", "383", "322", "605", "442", "172", "600", "270", "405", "322", "495", "359", "273", "379", "504", "385", "258", "282", "467", "559", "387", "615", "151", "473", "700", "369", "381", "378", "513", "70", "561", "442", "261", "445", "390", "453", "397", "349", "664"}
Dim Khulna() As String = {"180", "34", "439", "150", "109", "145", "295", "285", "184", "366", "142", "272", "514", "434", "155", "273", "363", "217", "54", "338", "378", "59", "90", "105", "345", "383", "0", "292", "444", "151", "211", "439", "113", "103", "211", "167", "378", "181", "302", "323", "90", "188", "227", "227", "319", "338", "454", "246", "200", "539", "141", "56", "177", "272", "417", "400", "59", "136", "394", "278", "472", "416", "273", "502"}
Dim Kishoreganj() As String = {"117", "290", "379", "360", "281", "317", "201", "79", "221", "305", "305", "159", "453", "342", "214", "212", "271", "99", "239", "132", "125", "277", "294", "281", "253", "322", "292", "0", "352", "256", "242", "347", "203", "253", "145", "294", "172", "130", "68", "251", "243", "115", "75", "214", "306", "65", "363", "221", "220", "447", "316", "297", "200", "260", "356", "308", "352", "213", "125", "138", "266", "210", "116", "411"}
Dim Kurigram() As String = {"348", "477", "661", "594", "515", "551", "151", "425", "460", "588", "341", "441", "736", "134", "388", "495", "120", "320", "473", "479", "323", "384", "528", "339", "157", "605", "444", "352", "0", "293", "482", "29", "438", "365", "339", "331", "519", "372", "329", "197", "418", "362", "368", "217", "309", "368", "111", "504", "276", "196", "550", "499", "357", "263", "639", "56", "456", "449", "339", "223", "613", "557", "256", "159"}
Dim Kushtia() As String = {"183", "184", "499", "301", "222", "258", "144", "268", "235", "425", "49", "279", "573", "283", "96", "332", "212", "184", "180", "322", "227", "92", "236", "46", "194", "442", "151", "256", "293", "0", "262", "288", "164", "73", "120", "58", "362", "194", "234", "172", "125", "199", "210", "76", "168", "273", "303", "297", "49", "388", "257", "207", "65", "122", "476", "249", "164", "188", "243", "127", "455", "400", "161", "351"}
Dim Lakshmipur() As String = {"137", "209", "229", "241", "162", "198", "331", "164", "43", "155", "284", "89", "303", "472", "171", "62", "402", "171", "157", "239", "313", "222", "176", "248", "384", "172", "211", "242", "482", "262", "0", "477", "98", "219", "200", "310", "279", "150", "257", "381", "189", "135", "160", "338", "430", "292", "493", "35", "312", "578", "197", "209", "198", "384", "206", "438", "270", "89", "322", "268", "373", "317", "227", "541"}
Dim Lalmonirhat() As String = {"343", "472", "656", "589", "510", "546", "146", "420", "455", "583", "336", "436", "731", "129", "383", "490", "115", "315", "468", "474", "318", "379", "523", "334", "152", "600", "439", "347", "29", "288", "477", "0", "433", "360", "334", "326", "514", "367", "324", "192", "413", "357", "363", "212", "304", "363", "106", "499", "271", "191", "545", "494", "352", "258", "634", "51", "451", "444", "334", "218", "608", "552", "251", "154"}
Dim Madaripur() As String = {"90", "110", "327", "143", "64", "100", "287", "195", "71", "253", "186", "138", "401", "428", "73", "160", "358", "127", "59", "248", "269", "124", "77", "149", "340", "270", "113", "203", "438", "164", "98", "433", "0", "121", "133", "212", "289", "91", "213", "337", "90", "98", "137", "240", "332", "249", "449", "133", "214", "534", "99", "111", "100", "286", "304", "395", "172", "23", "278", "224", "382", "326", "183", "497"}
Dim Magura() As String = {"150", "136", "461", "258", "179", "215", "217", "254", "192", "387", "65", "241", "536", "355", "52", "294", "285", "172", "105", "308", "278", "44", "193", "28", "267", "405", "103", "253", "365", "73", "219", "360", "121", "0", "108", "91", "348", "151", "247", "245", "50", "158", "197", "148", "241", "283", "376", "254", "122", "461", "214", "158", "75", "194", "439", "322", "115", "144", "294", "200", "442", "386", "192", "424"}
Dim Manikganj() As String = {"63", "200", "379", "270", "191", "227", "188", "148", "177", "305", "173", "159", "453", "329", "65", "212", "259", "64", "149", "202", "170", "152", "205", "137", "241", "322", "211", "145", "339", "120", "200", "334", "133", "108", "0", "178", "242", "89", "138", "239", "156", "80", "90", "202", "294", "174", "350", "221", "169", "435", "226", "208", "55", "248", "356", "296", "224", "156", "186", "125", "335", "280", "84", "398"}
Dim Meherpur() As String = {"240", "201", "552", "348", "270", "306", "182", "326", "283", "478", "26", "332", "626", "321", "143", "385", "250", "241", "197", "380", "266", "108", "256", "62", "232", "495", "167", "294", "331", "58", "310", "326", "212", "91", "178", "0", "420", "241", "272", "211", "142", "248", "268", "114", "206", "311", "342", "345", "88", "426", "305", "223", "122", "160", "529", "287", "180", "235", "281", "165", "513", "457", "199", "390"}
Dim Moulvibazar() As String = {"203", "376", "416", "446", "367", "403", "368", "116", "258", "342", "413", "196", "490", "509", "300", "249", "439", "208", "325", "63", "297", "362", "380", "376", "420", "359", "378", "172", "519", "362", "279", "514", "289", "348", "242", "420", "0", "216", "240", "418", "329", "201", "161", "381", "474", "237", "530", "258", "387", "614", "402", "383", "297", "427", "393", "475", "438", "299", "298", "305", "117", "58", "264", "578"}
Dim Munshiganj() As String = {"27", "179", "329", "248", "169", "205", "221", "123", "128", "256", "216", "109", "404", "362", "102", "163", "292", "61", "127", "176", "203", "165", "183", "179", "274", "273", "181", "130", "372", "194", "150", "367", "91", "151", "89", "241", "216", "0", "147", "271", "131", "15", "65", "234", "327", "182", "383", "172", "240", "468", "204", "186", "129", "280", "307", "328", "240", "101", "212", "158", "310", "254", "117", "431"}
Dim Mymensingh() As String = {"122", "300", "436", "370", "291", "327", "178", "147", "234", "362", "282", "216", "510", "320", "203", "269", "249", "93", "249", "200", "57", "291", "305", "280", "231", "379", "302", "68", "329", "234", "257", "324", "213", "247", "138", "272", "240", "147", "0", "229", "253", "137", "135", "192", "284", "39", "340", "278", "198", "425", "326", "308", "194", "238", "413", "286", "362", "223", "69", "115", "334", "278", "94", "388"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Naogaon() As String = {"247", "357", "561", "473", "395", "431", "50", "325", "359", "487", "221", "341", "635", "127",
"268", "394", "117", "219", "353", "378", "222", "264", "408", "219", "39", "504", "323", "251", "197", "172", "381", "192", "337",
"245", "239", "211", "418", "271", "229", "0", "298", "261", "267", "97", "98", "268", "142", "403", "156", "219", "430", "379", "237",
"78", "538", "153", "336", "348", "238", "122", "512", "456", "156", "182"}
Dim Narail() As String = {"130", "107", "442", "227", "148", "184", "269", "235", "162", "368", "116", "222", "516", "408", "100",
"275", "338", "167", "55", "289", "309", "34", "162", "79", "319", "385", "90", "243", "418", "125", "189", "413", "90", "50", "156",
"142", "329", "131", "253", "298", "0", "138", "177", "201", "293", "289", "429", "223", "175", "513", "183", "114", "122", "247",
"419", "374", "103", "114", "318", "252", "422", "366", "223", "477"}
Dim Narayanganj() As String = {"17", "186", "314", "255", "177", "213", "211", "107", "113", "240", "223", "94", "389", "352",
"109", "147", "282", "51", "134", "161", "193", "172", "190", "186", "264", "258", "188", "115", "362", "199", "135", "357", "98",
"158", "80", "248", "201", "15", "137", "261", "138", "0", "49", "224", "317", "172", "373", "157", "230", "458", "212", "193", "135",
"270", "292", "318", "247", "108", "202", "148", "294", "239", "107", "421"}
Dim Narsingdi() As String = {"51", "225", "339", "294", "215", "251", "217", "67", "137", "265", "262", "119", "413", "358",
"148", "172", "287", "56", "173", "121", "198", "211", "229", "225", "269", "282", "227", "75", "368", "210", "160", "363", "137",
"197", "90", "268", "161", "65", "135", "267", "177", "49", "0", "230", "322", "140", "378", "181", "236", "463", "250", "232", "146",
"276", "316", "324", "286", "147", "200", "153", "254", "198", "112", "426"}
Dim Natore() As String = {"210", "260", "524", "377", "298", "334", "68", "288", "311", "450", "124", "304", "598", "207", "171",
"357", "137", "182", "256", "341", "185", "167", "311", "122", "118", "467", "227", "214", "217", "76", "338", "212", "240", "148",
"202", "114", "381", "234", "192", "97", "201", "224", "230", "092", "231", "228", "366", "59", "312", "333", "282", "140", "46",
"501", "173", "239", "263", "201", "85", "475", "419", "119", "276"}
Dim Nawabganj() As String = {"302", "352", "616", "469", "390", "426", "160", "380", "403", "542", "217", "396", "690", "216",
"264", "449", "229", "274", "348", "433", "278", "259", "404", "214", "128", "559", "319", "306", "309", "168", "430", "304", "332",
"241", "294", "206", "474", "327", "284", "98", "293", "317", "322", "92", "0", "323", "230", "458", "151", "308", "425", "375", "232",
"46", "593", "265", "332", "356", "293", "177", "567", "511", "211", "271"}
Dim Netrakona() As String = {"158", "336", "444", "406", "327", "363", "217", "144", "270", "370", "321", "224", "518", "359",
"239", "277", "288", "129", "285", "197", "96", "327", "340", "319", "270", "387", "338", "65", "368", "273", "292", "363", "249",
"283", "174", "311", "237", "182", "39", "268", "289", "172", "140", "231", "323", "0", "379", "286", "237", "464", "362", "343",
"230", "277", "421", "325", "398", "259", "96", "154", "331", "275", "133", "427"}
Dim Nilphamari() As String = {"359", "488", "672", "604", "526", "562", "162", "436", "471", "598", "352", "452", "747", "57",
"399", "505", "131", "330", "483", "490", "334", "395", "539", "350", "103", "615", "454", "363", "111", "303", "493", "106", "449",
"376", "350", "342", "530", "383", "340", "142", "429", "373", "378", "228", "230", "379", "0", "514", "286", "68", "561", "510",
"368", "274", "649", "55", "467", "459", "349", "233", "623", "567", "267", "62"}
Dim Noakhali() As String = {"158", "244", "208", "276", "197", "233", "353", "143", "78", "134", "319", "67", "282", "494", "206",
"41", "423", "193", "192", "218", "335", "257", "210", "282", "405", "151", "246", "221", "504", "297", "35", "499", "133", "254",
"221", "345", "258", "172", "278", "403", "223", "157", "181", "366", "458", "286", "514", "0", "347", "599", "232", "244", "233",
"412", "185", "460", "305", "124", "344", "289", "352", "296", "248", "562"}
Dim Pabna() As String = {"216", "234", "530", "350", "272", "308", "127", "294", "285", "456", "98", "310", "604", "266", "145",
"363", "195", "188", "230", "347", "191", "141", "285", "96", "177", "473", "200", "220", "276", "49", "312", "271", "214", "122",
"169", "88", "387", "240", "198", "156", "175", "230", "236", "59", "151", "237", "286", "347", "0", "371", "307", "256", "114", "105",
"507", "232", "213", "237", "207", "91", "481", "425", "125", "335"}
Dim Panchagarh() As String = {"443", "572", "757", "689", "610", "646", "246", "521", "555", "683", "437", "537", "831", "94",
"484", "590", "216", "415", "568", "574", "418", "480", "624", "434", "181", "700", "539", "447", "196", "388", "578", "191", "534",
"461", "435", "426", "614", "468", "425", "219", "513", "458", "463", "312", "308", "464", "68", "599", "371", "0", "645", "595",
"453", "296", "734", "140", "552", "544", "434", "318", "708", "652", "352", "37"}
Dim Patuakhali() As String = {"204", "151", "426", "44", "41", "77", "401", "308", "170", "352", "279", "237", "500", "540", "166",
"259", "470", "241", "152", "362", "383", "200", "52", "242", "451", "369", "141", "316", "550", "257", "197", "545", "99", "214",
"226", "305", "402", "204", "326", "430", "183", "212", "250", "333", "425", "362", "561", "232", "307", "645", "0", "85", "193",
"379", "403", "506", "197", "122", "392", "337", "495", "440", "296", "609"}
Dim Pirojpur() As String = {"185", "66", "438", "96", "53", "89", "350", "290", "182", "364", "198", "249", "512", "489", "147",
"271", "419", "222", "59", "343", "364", "115", "33", "161", "401", "381", "56", "297", "499", "207", "209", "494", "111", "158",
"208", "223", "383", "186", "308", "379", "114", "193", "232", "282", "375", "343", "510", "244", "256", "595", "85", "0", "174",
"328", "415", "455", "112", "134", "373", "334", "477", "421", "278", "558"}
Dim Rajbari() As String = {"118", "167", "434", "236", "158", "194", "208", "204", "171", "360", "108", "214", "509", "347", "31",
"267", "277", "119", "116", "257", "226", "119", "171", "103", "259", "378", "177", "200", "357", "65", "198", "352", "100", "75",
"55", "122", "297", "129", "194", "237", "122", "135", "146", "140", "232", "230", "368", "233", "114", "453", "193", "174", "0",
"186", "412", "313", "190", "123", "241", "180", "391", "335", "139", "416"}
Dim Rajshahi() As String = {"256", "306", "569", "423", "344", "380", "114", "334", "357", "496", "170", "350", "644", "205",
"217", "403", "182", "228", "302", "387", "231", "213", "357", "168", "116", "513", "272", "260", "263", "122", "384", "258", "286",
"194", "248", "160", "427", "280", "238", "78", "247", "270", "276", "46", "46", "277", "274", "412", "105", "296", "379", "328",
"186", "0", "547", "219", "285", "309", "247", "131", "521", "465", "164", "260"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    Dim Rangamati() As String = {"293", "415", "74", "447", "368", "404", "488", "278", "249", "78", "504", "203", "195",
"629", "390", "144", "558", "328", "363", "353", "470", "453", "382", "467", "540", "70", "417", "356", "639", "476", "206",
"634", "304", "439", "356", "529", "393", "307", "413", "538", "419", "292", "316", "501", "593", "421", "649", "185", "507",
"734", "403", "415", "412", "547", "0", "595", "476", "295", "479", "424", "487", "431", "383", "698"}
    Dim Rangpur() As String = {"304", "433", "618", "550", "471", "507", "107", "382", "416", "544", "297", "398", "692", "78",
"345", "451", "76", "276", "429", "435", "279", "340", "485", "295", "113", "561", "400", "308", "56", "249", "438", "51",
"395", "322", "296", "287", "475", "328", "286", "153", "374", "318", "324", "173", "265", "325", "55", "460", "232", "140",
"506", "455", "313", "219", "595", "0", "412", "405", "295", "179", "569", "513", "213", "103"}
    Dim Satkhira() As String = {"240", "94", "499", "206", "164", "200", "307", "344", "243", "425", "154", "331", "574", "446",
"168", "332", "376", "287", "114", "398", "391", "72", "145", "118", "358", "442", "59", "352", "456", "164", "270", "451",
"172", "115", "224", "180", "438", "240", "362", "336", "103", "247", "286", "239", "332", "398", "467", "305", "213", "552",
"197", "112", "190", "285", "476", "412", "0", "195", "406", "291", "531", "476", "307", "515"}
    Dim Shariatpur() As String = {"101", "134", "317", "166", "87", "123", "298", "205", "62", "244", "209", "128", "392",
"439", "96", "151", "368", "138", "82", "259", "280", "147", "101", "173", "350", "261", "136", "213", "449", "188", "89",
"444", "23", "144", "156", "235", "299", "101", "223", "348", "114", "108", "147", "263", "356", "259", "459", "124", "237",
"544", "122", "134", "123", "309", "295", "405", "195", "0", "289", "234", "392", "337", "193", "507"}
    Dim Sherpur() As String = {"188", "366", "501", "435", "357", "393", "188", "204", "300", "427", "291", "281", "576",
"329", "250", "334", "258", "159", "314", "258", "16", "334", "370", "289", "240", "445", "394", "125", "339", "243", "322",
"334", "278", "294", "186", "281", "298", "212", "69", "238", "318", "202", "200", "201", "293", "96", "349", "344", "207",
"434", "392", "373", "241", "247", "479", "295", "406", "289", "0", "124", "391", "335", "103", "397"}
    Dim Sirajganj() As String = {"134", "311", "447", "381", "302", "338", "72", "211", "246", "373", "176", "227", "522",
"213", "190", "280", "142", "105", "260", "265", "109", "219", "316", "173", "124", "390", "278", "138", "223", "127", "268",
"218", "224", "200", "125", "165", "305", "158", "115", "122", "252", "148", "153", "85", "177", "154", "233", "289", "91",
"318", "337", "334", "180", "131", "424", "179", "291", "234", "124", "0", "398", "342", "42", "281"}
    Dim Sunamganj() As String = {"296", "470", "509", "539", "460", "496", "462", "209", "351", "436", "507", "290", "584",
"603", "393", "343", "532", "301", "418", "137", "391", "456", "474", "470", "514", "453", "472", "266", "613", "455", "373",
"608", "382", "442", "335", "513", "117", "310", "334", "512", "422", "294", "254", "475", "567", "331", "623", "352", "481",
"708", "495", "477", "391", "521", "487", "569", "531", "392", "391", "398", "0", "68", "357", "671"}
    Dim Sylhet() As String = {"241", "414", "454", "483", "405", "441", "406", "153", "295", "380", "451", "234", "528", "547",
"337", "287", "476", "246", "363", "81", "335", "400", "418", "414", "458", "397", "416", "210", "557", "400", "317", "552",
"326", "386", "280", "457", "58", "254", "278", "456", "366", "239", "198", "419", "511", "275", "567", "296", "425", "652",
"440", "421", "335", "465", "431", "513", "476", "337", "335", "342", "68", "0", "301", "616"}
    Dim Tangail() As String = {"92", "270", "406", "340", "261", "297", "105", "170", "204", "332", "209", "186", "480", "246",
"148", "239", "176", "64", "219", "223", "87", "236", "275", "207", "158", "349", "273", "116", "256", "161", "227", "251",
"183", "192", "84", "199", "264", "117", "94", "156", "223", "107", "112", "119", "211", "133", "267", "248", "125", "352",
"296", "278", "139", "164", "383", "213", "307", "193", "103", "42", "357", "301", "0", "315"}
    Dim Thakurgaon() As String = {"407", "536", "720", "652", "574", "610", "210", "484", "519", "646", "400", "500", "795",
"58", "447", "553", "179", "378", "532", "538", "382", "443", "587", "398", "144", "664", "502", "411", "159", "351", "541",
"154", "497", "424", "398", "390", "578", "431", "388", "182", "477", "421", "426", "276", "271", "427", "62", "562", "335",
"37", "609", "558", "416", "260", "698", "103", "515", "507", "397", "281", "671", "616", "315", "0"}
```

এবার ডিক্লায়র করবেন ইনডেক্স ও শহরগুলোর নাম:

```
Dim ProthomIndex As Integer : Dim DithioIndex As Integer 'declare globals for index value of listBox
    'load names of cities in an array in the memory.
    Dim NamesOfCity() As String = {"Dhaka", "Bagerhat", "Bandarban", "Barguna", "Barisal", "Bhola", "Bogra",
"Brahmanbaria", "Chandpur", "Chittagong", "Chuadanga", "Comilla", "Cox'sBaxar", _
                    "Dinajpur", "Faridpur", "Feni", "Gaibandha", "Gazipur", "Gopalganj", "Habiganj", "Jamalpur", "Jessore",
"Jhalokathi", "Jhenaidah", "Joypurhat", _
                    "Khagrachhari", "Khulna", "Kishoreganj", "Kurigram", "Kushtia", "Lakshmipur", "Lalmonirhat",
"Madaripur", "Magura", "Manikganj", "Meherpur", _
                    "Moulvibazar", "Munshiganj", "Mymensingh", "Naogaon", "Narail", "Narayanganj", "Narsingdi",
"Natore", "Nawabganj", "Netrakona", "Nilphamari", _
                    "Noakhali", "Pabna", "Panchagarh", "Patuakhali", "Pirojpur", "Rajbari", "Rajshahi", "Rangamati",
"Rangpur", "Satkhira", "Shariatpur", "Sherpur", _
                    "Sirajganj", "Sunamganj", "Sylhet", "Tangail", "Thakurgaon"}
```

ব্যস! কষ্ট ও সতর্কতাসহ উপরোক্ত তথ্যাদি আপনাকে টাইপ করতেই হবে। প্রোগ্রামারকে কখনো লেজি হতে নেই!

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

ডিক্লারেশন শেষে ফর্মের ক্লাসের ভেতর প্রথম সাবরুটিন হবে ফর্মের লোড ইভেন্ট। ফর্মে ডবল ক্লিক করলেই এই লোড ইভেন্ট সাবটি দেখাবে। এতে যা লিখবেন তাহলো:-

```
' This load sub populates two ListBoxes with the names of Cities.
 Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      Label1.Text = "" : Label5.Text = "" : Label6.Text = "" 'we don't need those names
      ' ==== dimension array for display in the list boxes. Remember, this sub is executed only once. ===
      Dim a As Integer ' a is loop control integer
      For a = 0 To 63
         ListBox1.Items.Add(NamesOfCity(a)) '0 through 63 = 64 dirstict names in the declared array.
      Next a
      For a = 0 To 63
         ListBox2.Items.Add(NamesOfCity(a)) '0 through 63 = 64 dirstict names in the declared array.
      Next a
 End Sub
```

উপরের কোড ব্লকের কাজ হলো উভয় লিস্টবাক্সে ৬৪টি জিলা শহরের নাম লিখা। এবার আমরা প্রস্তত হয়েছি ব্যবহারকারীর চাহিদানুযায়ী দূরত্ব প্রদর্শন করতে। এ লক্ষ্যর্জনে প্রয়োজন হবে নিচের দু'টি সাবরুটিন ও দু'টি ফাংশন।

প্রথম সাবরুটিন: লিস্ট বাক্সে ডবল ক্লিক করলেই এই সাবটি দেখাবে। এতে যা লিখবেন তাহলো:-

```
'whatever selected is displayed in the labels
Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged
      Label1.Text = ListBox1.SelectedItem 'get the selected text and display in label
      Label6.Text = "is: "
      Label5.Text = ""
      Label8.Text = "Equal: "
End Sub
```

আমরা জেনে নিলাম কোন্ শহর থেকে দূরত্ব জানার ইচ্ছা করছেন ব্যবহারকারী। এবার জানতে হবে কোন্ শহর পর্যন্ত দূরত্ব জানবেন তিনি। এছাড়া দ্বিতীয় এ তথ্যটি জানার সাথে সাথে আমরা চাই দূরত্বও প্রদর্শিত হবে- যা আমাদের প্রোগ্রামের মূল লক্ষ্য। দূরত্ব কিলোমিটার ও মাইলের এককে হতে হবে। এই কাজগুলোর জন্য দরকার নিচের দ্বিতীয় অপেক্ষাকৃত দীর্ঘ সাবরুটিনটি। এটাও লিস্ট বাক্সের (ListBox2) সিলেক্টেট ইনডেক্স পরিবর্তন ইভেন্ট সাবরুটিন। লিস্ট বাক্সে ডবল ক্লিক করলেই এটি প্রদর্শিত হবে। আপনি যা লিখবেন তাহলো:-

```
 'main display in this event
Private Sub ListBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ListBox2.SelectedIndexChanged
    Dim IndexA As Integer 'to get from index
    Dim IndexB As Integer 'to get to index
    Dim TextSel1 As String 'from city name
    Dim TextSel2 As String 'to city name

    If ListBox1.SelectedItem = "" Or ListBox2.SelectedItem = "" Then Exit Sub 'make sure selection is true
    IndexA = ListBox1.SelectedIndex 'get the index as number
    IndexB = ListBox2.SelectedIndex 'get the index as number
    TextSel1 = ListBox1.SelectedItem 'from city name
    TextSel2 = ListBox2.SelectedItem 'to city name
          'MsgBox(IndexA & " : " & IndexB)
    Label5.Text = TextSel2 'to city name into display
```

পরের পৃষ্ঠায় ... ...

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        '=== special case Dhaka. ref point
    If IndexA = 0 Then
        If IndexB = 0 Then 'Dhaka to Dhaka chosen
            Label6.Text = "is: " & 0 & " km"
            Label8.Text = "Equal: " & 0 & " Miles"
            Exit Sub
        End If
        Label6.Text = "is: " & FindDisanceZero(IndexB) & " km" 'If both condition results FALSE, this func. called
    End If
        '============= check from which city chosen ========
    Label6.Text = "is: " & ManyatOnce() & " km" 'call function and display result
        '==================== get distance in miles and display'
    Dim KMvalue As Integer = Mid(Label6.Text, 5, (Len(Label6.Text) - 7)) 'get km value to a variable as number
    Label8.Text = "Equal: " & Math.Truncate(KMvalue * 0.621371) & " M" 'convert to miles and display
    ListBox1.SelectedItem = ""
    ListBox2.SelectedItem = ""
End Sub
```

লাইন বাই লাইন ব্যাখ্যা: প্রায় প্রতিটি লাইনের পাশে (সবুজ রঙে) কমেন্ট করেছি বুঝার জন্য। এরপরও মনে করছি বুঝিয়ে দেওয়া আপনার জন্য সুবিধে হবে।

ক. প্রথম চার লাইনে চারটি ভেরিয়েবল ডিক্লায়ার করেছি। এর দু'টি ইন্টেজার ও দু'টি স্ট্রিং টাইপ। এগুলোতে যথাক্রমে লিস্ট বাক্সের ইনডেক্স সংখ্যা ও সিলেক্টেড টেক্সট সংরক্ষিত থাকবে।
খ. পঞ্চম লাইন দ্বারা নিশ্চিত করা হয়েছে, ব্যবহারকারী উভয় লিস্ট বাক্সে কোনো শহরের নাম সিলেক্ট করেছেন। না করলে প্রোগ্রাম বন্ধ হয়ে যাবে। এমতাবস্থায় সাবরুটিন এক্সেকিউশন বন্ধ করা জরুরী- যা এই লাইনটি নিশ্চিত করে।
ঘ. এর পরের চারটি লাইন দ্বারা প্রথম চার লাইনে ডিক্লায়ার করা ভেরিয়েবলগুলোতে ভ্যালু সংরক্ষণ করেছি। অর্থাৎ উভয় লিস্ট বাক্সের ইনডেক্ট সংখ্যা ও শহরের নাম ভেরিয়েবলে নিয়ে এসেছি। আগের পৃষ্ঠার শেষ লাইনটি দ্বারা আমরা দ্বিতীয় লিস্ট বাক্সের সিলেক্টেট টেক্সট তথা শহরের নাম লেবেলে এনে প্রদর্শন করেছি। এবার আমাদের কাজ হলো সঠিক দূরত্ব কিলোমিটার ও মাইল এককে প্রদর্শন করা।
ঙ. দূরত্ব প্রদর্শনের জন্য আমরা একই ইভেন্টই ব্যবহার করবো। অর্থাৎ ব্যবহারকারী দেখবেন দ্বিতীয় লিস্ট বাক্সে ক্লিক করলেই দূরত্ব প্রদর্শিত হচ্ছে। এ কাজের জন্য প্রথমত বিশেষ কেইস বিবেচনা করতে হবে- তাহলো ঢাকা থেকে ঢাকা। উভয়টি সিলেক্ট হলে রেজাল্ট হবে ০। আমরা এ কাজের জন্য দু'টি ইফ-এন্ড ইফ [If - End If] ব্লক ব্যবহার করে চেক করেছি। দ্বিতীয় ইফ-এন্ড ইফ প্রথমটির ভেতরে নেস্ট করা হয়েছে। অন্যকথায়, যদি উভয় ইনডেক্স = ০ হয় তাহলে বুঝতে হবে ঢাকা থেকে ঢাকা চয়েজ করা হয়েছে। আর এর ফলাফল ০। সুতরাং ডিসপ্লেতে ০ প্রদর্শন করে প্রোগ্রাম এখানেই থেমে যেতে হবে। এজন্যই লেবেল ৬ ও ৮-এ ০ প্রদর্শন করে Exit Sub কোড দ্বারা সাবরুটিন থেকে বেরিয়ে আসার নির্দেশ দেওয়া হয়েছে।
চ. পরের লাইনেই মূলত দূরত্ব প্রদর্শনের নির্দেশ দেওয়া হয়েছে। এতে বলা আছে: লেবেল৬ এর টেক্স হবে:
"is: " & ManyatOnce() & " km" - যার অর্থ, উভয়দিকে দু'টি শব্দ [is: , km] থাকবে এবং মাঝে লিখা হবে দূরত্ব যা আসবে [ManyatOnce()] ফাংশনের ফলাফল থেকে।
ছ. পরের দু'টি লাইনের কাজ হলো কিলোমিটার একককে পরিবর্তন করে মাইল এককে রূপান্তর ও লিস্ট বাক্স ৮-এ তা প্রদর্শন করা। সুতরাং ব্যবহারকারী কিলোমিটার ও মাইল এ উভয় এককে দূরত্ব জানতে পারবেন। সর্বশেষ দু'টি লাইন দ্বারা উভয় লিস্ট বাক্সের আইটেম সিলেকশন বাতিল করা হয়েছে। ফলে পরবর্তীতে ভুল ফলাফল আসার সম্ভাবনা আর থাকবে না।
জ. এবার শেষের ফাংশন সম্পর্কে কিছু বলার আছে। প্রথমে এটি কী তা [পরের পৃষ্ঠায়] দেখে নিই।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Function ManyatOnce() 'called from above procedure. Returns correct distance between cities.
    ProthomIndex = ListBox1.SelectedIndex : DithioIndex = ListBox2.SelectedIndex 'get index of selected index of listBox
    If ProthomIndex = 0 Then Return Dhaka(DithioIndex)
    If ProthomIndex = 1 Then Return Bagerhat(DithioIndex)
    If ProthomIndex = 2 Then Return Bandarban(DithioIndex)
    If ProthomIndex = 3 Then Return Barguna(DithioIndex)
    If ProthomIndex = 4 Then Return Barisal(DithioIndex)
    If ProthomIndex = 5 Then Return Bhola(DithioIndex)
    If ProthomIndex = 6 Then Return Bogra(DithioIndex)
    If ProthomIndex = 7 Then Return Brahmanbaria(DithioIndex)
    If ProthomIndex = 8 Then Return Chandpur(DithioIndex)
    If ProthomIndex = 9 Then Return Chittagong(DithioIndex)
    If ProthomIndex = 10 Then Return Chuadanga(DithioIndex)
    If ProthomIndex = 11 Then Return Comilla(DithioIndex)
    If ProthomIndex = 12 Then Return coxsBazar(DithioIndex)
    If ProthomIndex = 13 Then Return Dinajpur(DithioIndex)
    If ProthomIndex = 14 Then Return Faridpur(DithioIndex)
    If ProthomIndex = 15 Then Return Feni(DithioIndex)
    If ProthomIndex = 16 Then Return Gaibandha(DithioIndex)
    If ProthomIndex = 17 Then Return Gazipur(DithioIndex)
    If ProthomIndex = 18 Then Return Gopalganj(DithioIndex)
    If ProthomIndex = 19 Then Return Habiganj(DithioIndex)
    If ProthomIndex = 20 Then Return Jamalpur(DithioIndex)
    If ProthomIndex = 21 Then Return Jessore(DithioIndex)
    If ProthomIndex = 22 Then Return Jhalokathi(DithioIndex)
    If ProthomIndex = 23 Then Return Jhenaidah(DithioIndex)
    If ProthomIndex = 24 Then Return Joypurhat(DithioIndex)
    If ProthomIndex = 25 Then Return Khagrachhari(DithioIndex)
    If ProthomIndex = 26 Then Return Khulna(DithioIndex)
    If ProthomIndex = 27 Then Return Kishoreganj(DithioIndex)
    If ProthomIndex = 28 Then Return Kurigram(DithioIndex)
    If ProthomIndex = 29 Then Return Kushtia(DithioIndex)
    If ProthomIndex = 30 Then Return Lakshmipur(DithioIndex)
    If ProthomIndex = 31 Then Return Lalmonirhat(DithioIndex)
    If ProthomIndex = 32 Then Return Madaripur(DithioIndex)
    If ProthomIndex = 33 Then Return Magura(DithioIndex)
    If ProthomIndex = 34 Then Return Manikganj(DithioIndex)
    If ProthomIndex = 35 Then Return Meherpur(DithioIndex)
    If ProthomIndex = 36 Then Return Moulvibazar(DithioIndex)
    If ProthomIndex = 37 Then Return Munshiganj(DithioIndex)
    If ProthomIndex = 38 Then Return Mymensingh(DithioIndex)
    If ProthomIndex = 39 Then Return Naogaon(DithioIndex)
    If ProthomIndex = 40 Then Return Narail(DithioIndex)
    If ProthomIndex = 41 Then Return Narayanganj(DithioIndex)
    If ProthomIndex = 42 Then Return Narsingdi(DithioIndex)
    If ProthomIndex = 43 Then Return Natore(DithioIndex)
    If ProthomIndex = 44 Then Return Nawabganj(DithioIndex)
    If ProthomIndex = 45 Then Return Netrakona(DithioIndex)
    If ProthomIndex = 46 Then Return Nilphamari(DithioIndex)
    If ProthomIndex = 47 Then Return Noakhali(DithioIndex)
    If ProthomIndex = 48 Then Return Pabna(DithioIndex)
    If ProthomIndex = 49 Then Return Panchagarh(DithioIndex)
    If ProthomIndex = 50 Then Return Patuakhali(DithioIndex)
    If ProthomIndex = 51 Then Return Pirojpur(DithioIndex)
    If ProthomIndex = 52 Then Return Rajbari(DithioIndex)
    If ProthomIndex = 53 Then Return Rajshahi(DithioIndex)
    If ProthomIndex = 54 Then Return Rangamati(DithioIndex)
    If ProthomIndex = 55 Then Return Rangpur(DithioIndex)
    If ProthomIndex = 56 Then Return Satkhira(DithioIndex)
    If ProthomIndex = 57 Then Return Shariatpur(DithioIndex)
    If ProthomIndex = 58 Then Return Sherpur(DithioIndex)
    If ProthomIndex = 59 Then Return Sirajganj(DithioIndex)
    If ProthomIndex = 60 Then Return Sunamganj(DithioIndex)
    If ProthomIndex = 61 Then Return Sylhet(DithioIndex)
    If ProthomIndex = 62 Then Return Tangail(DithioIndex)
    If ProthomIndex = 63 Then Return Thakurgaon(DithioIndex)
    Return 0
  End Function
End Class
```

ফাংশনটি বেশ দীর্ঘ - তাই না? আসলে এটা টাইপ করা তেমন কঠিন কিছু না। আপনি একটি টাইপ করে কপি-পেস্ট করবেন। এরপর শুধুমাত্র দু'টি শব্দ বদলাতে হবে। প্রথমত, ১ থেকে ৬৩ সংখ্যা হবে এবং দ্বিতীয়ত, ঢাকা থেকে ঠাকুরগাঁও লেখা থাকবে। তবে আপনাকে মনে রাখতে হবে:

(ক) ৬৬-টি লাইনের মধ্যে যে কোনো একটি মাত্র ভ্যালু রিটার্ন করবে। অর্থাৎ ব্যবহারকারী প্রথম লিস্ট বাক্সে যে শহরের ইনডেক্সে ক্লিক করবেন তা-ই নির্ধারণ করবে কোন্ ভ্যালু ফাংশন ফেরত দেবে।

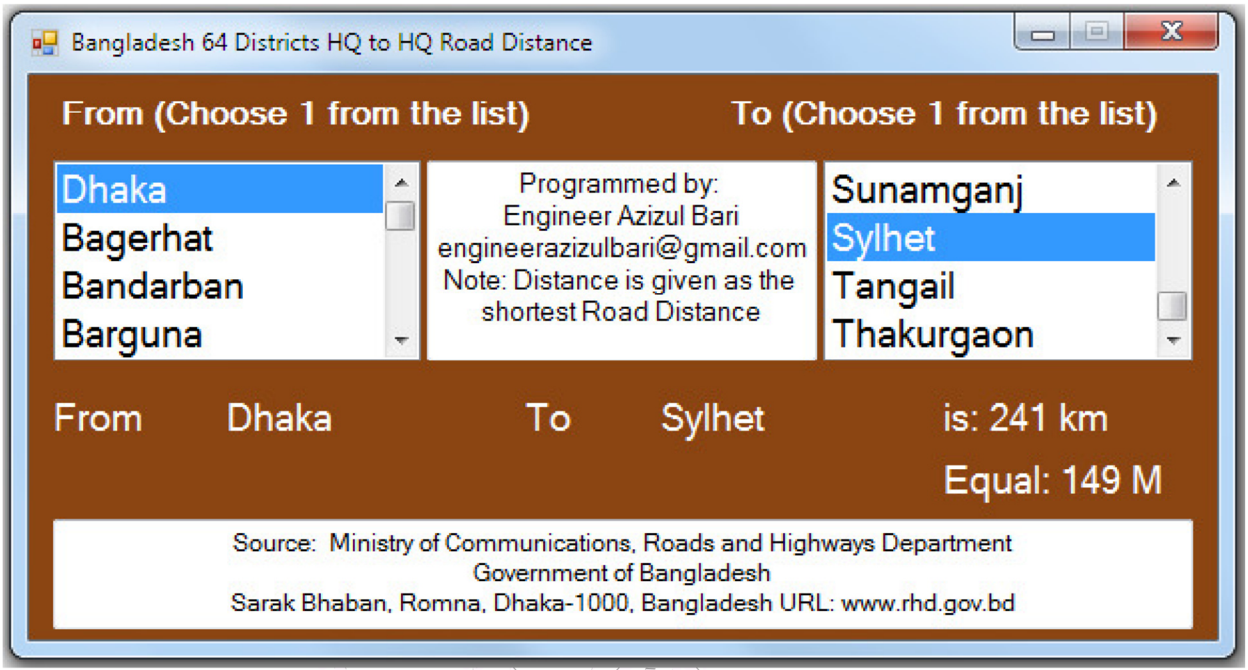
(খ) ফেরৎ ভ্যালু আসবে ফর্মের প্রথমে ডিক্লায়ার করা শহরের নামের এ্যারে ভেরিয়েবল থেকে। সুতরাং এই নামগুলোর বানান সঠিক থাকা দরকার। আমি টাইপ করার সময় ডিক্লারেশন লেখাটুকু কপি করে কাছে নিয়ে এসেছিলাম। এরপর একটা একটা করে সবগুলো কাট-পেস্ট করে বসিয়েছি। ব্যস্! ভুল হওয়ার সম্ভাবনা আর থাকে নি।

যাক, ব্যবহারযোগ্য একটি প্রোগ্রাম আমরা সৃষ্টি করতে সমর্থ হয়েছি। এখন বাকী থাকলো টেস্ট করার বিষয়টি। সতুরাং প্রোগ্রাম রান করুন এবং একটা একটা করে শহরের নামে ক্লিক করে দেখুন দূরত্ব সঠিকমতো আসছে কি না। সব ঠিকমতো কোড করা হয়ে থাকলে অবশ্যই সঠিক ফলাফল আসবে। আর কোনো সমস্যা হলে খুঁজে দেখতে হবে কোথায় ভুল-ত্রুটি হয়েছে। আর এভাবে ভুল-সংশোধনকেই বলে ডিবাগিং [Debugging]।

পরের পৃষ্ঠায় প্রোগ্রাম চলাকালের একটি চিত্র দিয়েছি। একটু দেখে নিন।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

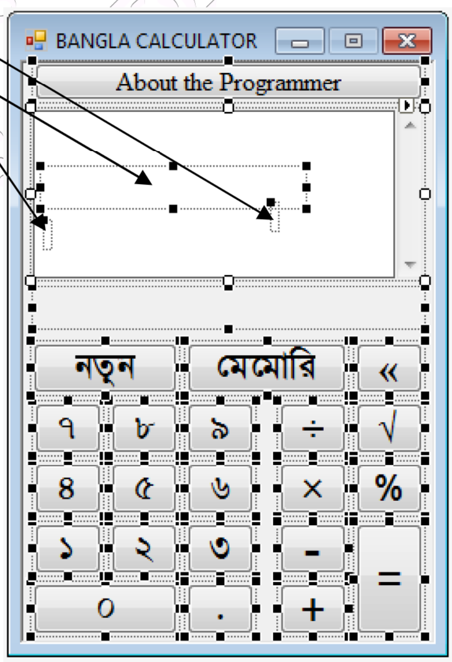


Application 1 -চলাকালীন ঢাকা থেকে সিলেটের দূরত্ব কিলোমিটার ও মাইল এককে প্রদর্শিত হচ্ছে।

Application 2 : Bangla Calculator - বাংলা ক্যালকুলেটর

উদ্দেশ্য: প্রশ্ন থাকতে পারে, বাংলা ক্যালকুলেটর- কেনো? জবাব- কেনো না? তবে আসল জবাব হলো- আমরা তো শিখতে চাই! ক্যালকুলেটর কোডিং কিভাবে করতে হয় তা জানতে চাই। আর ডিসপ্লে বাংলা সংখ্যায় হলে তো ভলোই হলো! সুতরাং আসুন, গড়ে তুলি পিসি'র জন্য একটি উন্নত বাংলা ক্যালকুলেটর এ্যাপলিকেশন।

ডিজাইন: ডানের ডিজাইনটি দেখুন। আমাদের ক্যালকুলেটরে সাধারণ পাটিগণিতের চারটি ফাংশন [+, -, ÷, x] ছাড়াও থাকবে পার্সেন্ট (%) এবং বর্গমূল (√)।



উপরে আছে প্রোগ্রামার সম্পর্কে তথ্য প্রদর্শনের একটি বোতাম [About the Programmer]। এর পরে আছে মূল প্রদর্শন জানালা [TextBox1]। এ বাক্সেই সংখ্যা প্রবেশকরণ ও প্রদর্শন ক্রিয়া সম্পাদিত হবে। এটাই হলো, মূল ডিসপ্লে। আমরা তিনটি লুকানো [label2, label3, label4] লেবেল দেখছি। এগুলোতে কিছু তথ্য সংরক্ষিত থাকবে যা ব্যবহারকারী দেখার প্রয়োজন নেই। সুতরাং এদের ভিজিবল প্রোপার্টি হবে: False। ডিসপ্লের একটু নিচে লম্বাটে একটি লেবেল [label1] আছে। এটা দ্বারা ব্যবহারকারীকে কিছু তথ্য প্রদর্শন করা হবে। বাকী সবই বোতাম। উপরের সারির তিনটি বোতামের কাজ যথাক্রমে: নতুন হিসাব, সংরক্ষিত সংখ্যা দেখা ও ডিসপ্লেতে সর্বশেষ সংখ্যা মেটানো। বাকি সব বোতামের কাজ কি শিক্ষার্থীরা সহজেই অনুমান করতে পারছেন। সুতরাং আর বেশি বলছি না। তবে মনে রাখবেন, বোতামের টেক্সট [Text] প্রোপার্টিতে যা দেখা যাচ্ছে তা ঢুকাতে হবে। অর্থাৎ ১, ২, ০, ., =, %, √ ইত্যাদি হবে বোতামগুলোর টেক্সট প্রোপার্টি।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

ওহ! আরেক কথা- ÷ লিখতে বাংলায় বিজয় কিবোর্ডে **স্ট** লিখে টাইমস্ ফন্টে পরিবর্তন করুন। √ লিখতে ম্যানু থেকে সিম্বুল ইনসার্ট করতে পারেন। আর « এটার জন্য কিবোর্ডের মাধ্যমে ্র -র-ফলা প্রবেশ করে টাইমস্ ফন্টে পরিবর্তন করলেই হবে।

যাক, এই হচ্ছে ডিজাইন। চাইলে আপনি অন্যভাবেও করতে পারেন- কিছু রঙ যুক্ত করে আকর্ষণীয় করুন কিংবা ক্যালকুলেটরের সাইজ বৃদ্ধি করুন ইত্যাদি। তবে মনে রাখবেন যে অবজেক্টগুলো ফর্মে আছে তা সবই যেনো থাকে। কারণ এগুলোর নাম নিয়ে আমরা কোড করেছি।

কোডিং: ক্যালকুলেটরের কোডিং আসলে তেমন কঠিন নয়! সত্যিই বলছি। তবে হ্যাঁ, সহজ করতে যেয়ে কিছু চিন্তা–ভাবনা তো অবশ্যই করতে হবে এবং কোডিং অভিজ্ঞতা কাজে লাগানো দরকার। আসুন তাহলে কোডগুলো দেখে নিই।

আমাদের যাবতীয় কোডই শুধুমাত্র ফর্মের ক্লাস ইভেন্ট সাবরুটিনে লিখিত হয়েছে। এবার একে একে ক্লাসের ভেতরের সাবরুটিন ও ফাংশনগুলোর ওপর বিস্তারিত আলোচনার প্রয়াস পাচ্ছি।

```
Public Class Form1
    'this function tests if the display empty or not.
    'if empty returns true, otherwise false
    Function DisplayNotEmpty()
        If TextBox1.Text = "" Then
            Return True
        End If
        Return False
    End Function
```

উপরের ফাংশনের কাজ হলো মূল ডিসপ্লে বা প্রদর্শন টেক্স বাক্সে [TextBox1] সংখ্যা লিখা আছে কি না তা পরীক্ষা করে ফলাফল [True or False] রিটার্ন করা। এ ফাংশন আমরা প্রয়োজনে অন্যান্য সাবরুটিন থেকে কল করবো।

```
 'this sub is for '1-9, . and 0' entry to TextBox1 (the Display) directly
  Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
      e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -
1)))
      ' And (sender.Text.IndexOf(".") = -1)) -this ensures . isn't entered more then once.
    End Sub
```

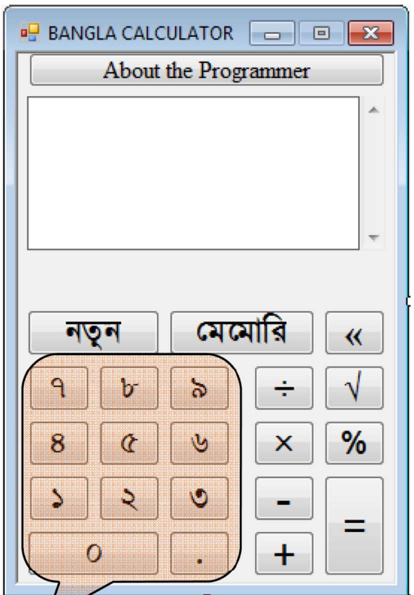
উপরের এ দ্বিতীয় সাবটি সম্পর্কে ইতোমধ্যে বলেছি। এটার কাজ হলো টেক্স বাক্সে শুধুমাত্র আস্ত কিংবা দশমিক সংখ্যা ঢুকানোর নিশ্চয়তা দেওয়া। অন্য কিছুই টাইপ করে টেক্স বাক্সে ঢুকানো যাবে না। এ কোডটি টেক্স বাক্সের KeyPress ইভেন্ট হ্যান্ডলারে দিতে হবে। এই হ্যান্ডলার পেতে প্রথামে কোড জানালার বায়ের লিস্টের সর্বশেষ আইটেম [TextBox1] চয়েজ করুন। এবার ডানের লিস্ট খুলে চয়েজ করুন কী প্রেস [KeyPress] আইটেমটি। ব্যস্! উপরোক্ত হ্যান্ডলার সাবটি এমনিতেই এসে যাবে। আপনাকে এর ভেতরে [উপরে দেখানো] একটি মাত্র লাইন লিখতে হবে।

এর পরে মোট ১১টি সাবরুটিন আছে যাদের কাজ একই ধরনের। সংখ্যা ১ থেকে ৯, ০ এবং . [দশমিক বিন্দু] টেক্স বাক্সে ঢুকানোর জন্য মোট ১১টি বোতাম আছে। ওগুলোতে ক্লিক করলেই সংশ্লিষ্ট সংখ্যা [বা একটি দশমিক বিন্দু] বাক্সে প্রদর্শিত হবে। পরের পৃষ্ঠায় এই ১১টি কোড ব্লক একত্রে দিয়েছি। দেখে দেখে টাইপ করে নিন। মনে রাখবেন, ক্লিক ইভেন্ট হ্যান্ডলার আসবে বোতামগুলো ডবল ক্লিক করার মাধ্যমে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
'this sub is for '.' entry
  Private Sub ButPoint_Click(sender As Object, e As EventArgs) Handles ButPoint.Click
     If InStr(TextBox1.Text, ".") > 0 Then Exit Sub 'make sure '.' not entered more then once.
     TextBox1.Text = TextBox1.Text & "." 'add the point after existing text
  End Sub
  'But0 - But9 codes
  Private Sub But0_Click(sender As Object, e As EventArgs) Handles But0.Click
     TextBox1.Text = TextBox1.Text & "0" 'must keep the previous number
  End Sub
  Private Sub But1_Click(sender As Object, e As EventArgs) Handles But1.Click
     TextBox1.Text = TextBox1.Text & "1"
  End Sub
  Private Sub But2_Click(sender As Object, e As EventArgs) Handles But2.Click
     TextBox1.Text = TextBox1.Text & "2"
  End Sub
  Private Sub But3_Click(sender As Object, e As EventArgs) Handles But3.Click
     TextBox1.Text = TextBox1.Text & "3"
  End Sub
  Private Sub But4_Click(sender As Object, e As EventArgs) Handles But4.Click
     TextBox1.Text = TextBox1.Text & "4"
  End Sub
  Private Sub But5_Click(sender As Object, e As EventArgs) Handles But5.Click
     TextBox1.Text = TextBox1.Text & "5"
  End Sub
  Private Sub But6_Click(sender As Object, e As EventArgs) Handles But6.Click
     TextBox1.Text = TextBox1.Text & "6"
  End Sub
  Private Sub But7_Click(sender As Object, e As EventArgs) Handles But7.Click
     TextBox1.Text = TextBox1.Text & "7"
  End Sub
  Private Sub But8_Click(sender As Object, e As EventArgs) Handles But8.Click
     TextBox1.Text = TextBox1.Text & "8"
  End Sub
  Private Sub But9_Click(sender As Object, e As EventArgs) Handles But9.Click
     TextBox1.Text = TextBox1.Text & "9"
  End Sub
```



এই ১১ বোতামের কোড উপরে দেখানো হয়েছে।

লক্ষ করুন: ButPoint বোতামের কোডের প্রথম লাইন দ্বারা নিশ্চিত করা হয়েছে, দশমিক বিন্দু [.] যাতে একবারই ঢুকানো সম্ভব হয়।

প্রত্যেক সাবরুটিনে বলা হয়েছে:
TextBox1.Text = TextBox1.Text & ...

এভাবে বলার কারণ কি জানেন? আমরা চাইনা আগের সংখ্যা মুছে যাবে। এটা নিশ্চিত হয়েছে TextBox1.Text & বলার মাধ্যমে। এরপর এসেছে সংশ্লিষ্ট সংখ্যাটি ঢুকানোর নির্দেশ। সুতরাং প্রত্যেকবার প্রথমে টেক্স বাক্সে যা আছে তা সংরক্ষণ হবে এবং এর সাথে ঢুকানো সংখ্যাটি যুক্ত হয়ে প্রদর্শিত হবে।

মনে রাখবের বোতামের মধ্যে ডবল ক্লিক করলেই ক্লিক ইভেন্ট সাবটি আসবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

পরবর্তী কোড হলো, ButNew নামক বোতামের ক্লিক ইভেন্ট। আমরা বাংলায় এর নামকরণ করেছি 'নতুন'। এটার কাজ হলো সব মুছে নতুন হিসাব করার জন্য ক্যালকুলেটরকে প্রস্তুত করা। তবে মেমোরি মুছে যাবে না।

```
'clears all for new calculation but not the memory value kept in label2
 Private Sub ButNew_Click(sender As Object, e As EventArgs) Handles ButNew.Click
      Label3.Text = "" : TextBox1.Text = ""
      Label1.Text = "" : Label4.Text = ""
 End Sub
```

লক্ষ করুন: Label3 -তে থাকবে +, - ইত্যাদি সংকেত চিহ্ন। এটা দ্বারা বুঝা যাবে ইতোমধ্যে ব্যবহারকারী প্রথম সংখ্যা ঢুকিয়ে সংকেত বোতামে ক্লিক করে নিয়েছেন কি না। কারণ, ক্লিক করে নিয়ে থাকলে তো আর পুনরায় ঢুকানোর প্রয়োজন নেই। TextBox1 -এ থাকবে প্রদর্শিত সংখ্যা। Label1 -এ থাকবে হিসাবের তথ্যাদি [এটাই হলো বোতামগুলোর উপরস্থ লম্বাটে লেবেলটি]। সবশেষে Label4-এ থকাবে ক্যালকুলেশনের জন্য ব্যবহৃত সংকেত।

```
 'gets the value from memory to Display
   Private Sub ButMem_Click(sender As Object, e As EventArgs) Handles ButMem.Click
      If Label2.Text <> "" Then TextBox1.Text = Label2.Text 'make sure it contains value before import
   End Sub
```

উপরের সাবের কাজ হলো [Label2 -তে] সংরক্ষিত মেমোরির সংখ্যা ডিসপ্লে ফিল্ডে নিয়ে আসা। লক্ষ করুন, প্রথমে টেস্ট এসেছে মেমোরিতে কিছু আছে কি না- থাকলেই তো আনা যাবে। এর পরের ৫টি সাবের কাজ প্রায় একই। এগুলো দ্বারা +, - ইত্যাদি সংকেত সংশ্লিষ্ট লেবেলে সংরক্ষণ করা হয়। লক্ষ করুন প্রতিটি সাবই `DisplayNotEmpty` নামক একটি সাবরুটিন ও `FirstValue` নামক একটি ফাংশন কল করে। প্রথম সাবটির কথা আগেই বলেছি। এর কাজ হচ্ছে ডিসপ্লেতে সংখ্যা আছে কি না তা চেক করা। আর ফাংশনটির কাজ হলো ডিসপ্লে থেকে সংখ্যা কপি করে লুকানো লেবেল [Label3 -তে] নিয়ে যাওয়া এবং ডিসপ্লে মুছে দেওয়া। আমরা এ সম্পর্কে আরো কিছু ব্যাখ্যা একটু পরই তুলে ধরবো। প্রথমে ৫টি সাব লিখে দিচ্ছি।

```
'These 5 subs entries operand signs in label4 and label1. Calls firstValue sub
   'to entry value into hidden Label3 and clears display
   'Plus Button
Private Sub ButPlus_Click(sender As Object, e As EventArgs) Handles ButPlus.Click
      If Label3.Text <> "" Then Exit Sub 'don't do anything if the sign is already there
      If DisplayNotEmpty() = False Then 'check there is number in the display
         Label1.Text = TextBox1.Text & " +" : Label4.Text = "+" 'sign in lavel4 is used to calculate
         FirstValue() 'sub call
      End If
End Sub
   'Minus Button
Private Sub ButMinus_Click(sender As Object, e As EventArgs) Handles ButMinus.Click
      If Label3.Text <> "" Then Exit Sub
      If DisplayNotEmpty() = False Then
         Label1.Text = TextBox1.Text & " -" : Label4.Text = "-"
         FirstValue()
      End If
End Sub
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'Times Button
Private Sub ButTimes_Click(sender As Object, e As EventArgs) Handles ButTimes.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " ¸Y" : Label4.Text = "x"
            FirstValue()
        End If
End Sub
    'Divide Button
Private Sub ButDivide_Click(sender As Object, e As EventArgs) Handles ButDivide.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " /" : Label4.Text = "÷"
            FirstValue()
        End If
End Sub
    'percentage Button
    Private Sub ButPercent_Click(sender As Object, e As EventArgs) Handles ButPercent.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " % " : Label4.Text = "%"
            FirstValue()
        End If
End Sub
```

উপরের প্রতিটি সাবের কাজ যেহেতু একই ধরনের তাই একটির ওপর ব্যাখ্যা তুলে ধরাই যথেষ্ট হবে। সাবের প্রথম লাইন দ্বারা নিশ্চিত করা হচ্ছে ইতোমধ্যে ব্যবহারকারী কোনো সংকেতে ক্লিক করে নেন নি। যদি করে থাকেন তাহলে তো লেবেলে [Label3 -তে] সংকেত ঢুকানো হয়েই গেলো। সুতরাং [Exit Sub -দ্বারা] সাব থেকে বের হতে হবে।

দ্বিতীয় চেকটি হলো ডিসপ্লেতে সংখ্যা আছে কি না তা জানা। সংখ্যা থাকলেই [ফাংশনের ফলাফল] পরের লাইনে যেয়ে আমরা প্রথমে লেবেল১-এ ব্যবহারকারীকে কিছু তথ্য দিচ্ছি এবং দ্বিতীয়ত গোপন লেবেল৪-এ সংকেতটি সংরক্ষণ করছি। এরপর ফাস্টভ্যালু নামক সাবরুটিন কল করেছি। যার কাজ হলো লেবেল৩-তে ডিসপ্লের সংখ্যা নিয়ে সংরক্ষণ করা ও ডিসপ্লে মুছে দেওয়া। এই কাজগুলো উপরের ৫টি সাব দ্বারা সম্পাদিত হয়। এবার আমরা ‘স্কোয়ার রুট’ [বর্গমূল] ক্যালকুলেশনের বিশেষ সাবটির ওপর তথ্যাদি তুলে ধরছি।

```
'Square Root Calculation and display sub
Private Sub ButSqr_Click(sender As Object, e As EventArgs) Handles ButSqr.Click
        If TextBox1.Text <> "" And Label1.Text = "" Then 'make sure number exists
            Dim SquareRoot As Double = Math.Sqrt(TextBox1.Text) 'get sqr into variable
            Label1.Text = TextBox1.Text & " Gi eM©g~j" 'bangla info display
            TextBox1.Text = Math.Round(SquareRoot, 8) 'put result into display
            Label2.Text = SquareRoot 'put result into memory
        End If
End Sub
```

বুঝতেই পারছেন, বর্গমূল বের করা অন্যান্য সকল কাজ থেকে ভিন্ন। এর কারণ হলো, বর্গমূলের জন্য একটি মাত্র সংখ্যার প্রয়োজন। অপরদিকে +, –, ÷, x ও % বের করতে দু’টি সংখ্যা থাকা জরুরী। সুতরাং ওগুলোর ক্যালকুলেশনও হবে একই ধরনের।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

উপরের সাবের প্রথম লাইনের কাজ হলো ডিসপ্লে ফিল্ডে একটি সংখ্যা থাকা নিশ্চিত করা। এই টেস্ট সত্যি ফলাফল রিটার্ন না করলে সাব থেকে বেরিয়ে যেতে হবে। যাক, টেস্ট সফল হওয়ার নিশ্চয়তা লাভের পর আমরা বর্গমূল বের করার কাজে যেতে পরি- যা পরবর্তী চারটি লাইনের ক্রিয়া। প্রথমেই আমরা একটি ভেরিয়েবল ডিক্লায়ার করে তাতে সংরক্ষণ করেছি সংখ্যাটির বর্গমূল। লক্ষ করুন, ভেরিয়েবল হবে Double Type -এটা মূলত দশমিক সংখ্যা টাইপ। এছাড়া বর্গমূল বের করতে আমরা ব্যবহার করেছি, সিস্টেম ফাংশন Math.Sqrt(Number) । এই ফাংশন একটি মাত্র ফ্লোটিং পয়েন্ট [দশমিক] সংখ্যা প্যারামিটার গ্রহণ করে। আমাদের ক্ষেত্রে তাহলো ডিসপ্লেতে লিখিত সংখ্যাটি। পরের লাইনের কোড দ্বারা লেবেল-১-এ বাংলায় কিছু তথ্য দেওয়া হয়েছে। লক্ষ করুন, প্রদর্শিত তথ্যাদি হলো টেক্সট বাক্সের সংখ্যা ও বাংলায় এ কথাটি: " এর বর্গমূল"। এর পরের লাইন দ্বারা ভেরিয়েবলে সংরক্ষিত বর্গমূল সংখ্যা [৮টি দশমিক স্থান পর্যন্ত] প্রদর্শন করেছি ডিসপ্লে ফিল্ডে। এখানেও একটি সিস্টেম ফাংশন কাজে লাগিয়েছি: Math.Round (SquareRoot, 8)। এই ফাংশন একটি ফ্লোটিং পয়েন্ট সংখ্যা ও আরেকটি ইন্টেজার সংখ্যা প্যারামিটার হিসাবে গ্রহণ করে। দ্বিতীয়টি দ্বারা ক'টি পর্যন্ত ডেসিমেল প্লেইসেজ্ থাকবে তা নির্দিষ্ট হয়। সাবরুটিনের সর্বশেষ লাইনের কাজ হচ্ছে, ফলাফল [বর্গমূল সংখ্যা] লুকানো লেবেল-২ এ মেমোরি হিসাবে সংরক্ষণ করা।

```
 'this sub clears last charater from the display
Private Sub ButClear_Click(sender As Object, e As EventArgs) Handles ButClear.Click
      If TextBox1.Text = "" Then Exit Sub 'make sure there is text to be deleted
      TextBox1.Text = Mid(TextBox1.Text, 1, (Len(TextBox1.Text) - 1)) 'collects all text except last one
      'and displays
End Sub
```

উপরের এ সাবটির কাজ হলো ডিসপ্লেতে প্রদর্শিত সর্বশেষ ডিজিটটি মুছে ফেলা। প্রথমে ডিসপ্লেতে সংখ্যা আছে তা নিশ্চিত করে এ কাজটি করা হয়েছে। লক্ষ করুন, Mid -হলো একটি সিস্টেম ফাংশন। এর দ্বারা নির্দিষ্ট পরিমাণ লেখা সংগ্রহ করা যায়। প্রথম (১) থেকে শেষেরটির আগেরটি পর্যন্ত আমরা সংগ্রহ করে পুনরায় ডিসপ্লেতে প্রদর্শন করছি এই ফাংশন ব্যবহার করে। Mid -ফাংশন বেশ ক'টি প্যারামিটার গ্রহণ করে। এখানে আমরা তিনটি প্যারামিটার দিয়েছি। প্রথমটি [TextBox1.Text] হলো ডসপ্লের পুরো লেখাটুকু। দ্বিতীয়টি হলে লেখার কোথায় থেকে সংগ্রহ শুরু হবে- সে ক্যারাক্টার পজিশন সংখ্যা [এখানে ১]। তৃতীয়টি হলো কী পরিমাণ লেখা সংগ্রহ করা হবে- তা নির্দিষ্টকরণ, এখানে শেষ ক্যারাক্টার পজিশনের আগেরটা পর্যন্ত। এভাবে করায় ব্যবহারকারী দেখবেন, শেষের সংখ্যাটি মুছে গেছে।

এবার আমরা ক্যালকুলেটর কোডের মূল সাবরুটিনটি নিয়ে বিস্তারিত আলোচনা করবো। এ সাবরুটিনটিই হলো পুরো প্রোগ্রামের মগজ। ইতোমধ্যে যা হয়েছে [উপরের সাব বাদে] তা ছিলো প্রস্তুতি গ্রহণ ক্রিয়া। একটি ক্যালকুলেটরে সর্বদাই সমান চিহ্ন [=] থাকবে। এ চিহ্নটি দ্বারাই কাঙ্ক্ষিত ফলাফল হিসাব করে ডিসপ্লেতে প্রদর্শিত হয়। সুতরাং আসুন, আমাদের বাংলা ক্যালকুলেটরের সমান চিহ্নের বোতামের ক্লিক ইভেন্ট সাবরুটিন কিভাবে কাজ করে তা তলিয়ে দেখি। প্রথমেই আমরা পুরে সাবরুটিনটি [এ পৃষ্ঠা ও পরের পৃষ্ঠায়] তুলে ধরছি।

```
 'calculate value and display
Private Sub ButEqual_Click(sender As Object, e As EventArgs) Handles ButEqual.Click
    If CheckValidity() = False Then Exit Sub 'check if all in order
    Dim CalValue As Double 'declare variable to hold result
    If Label4.Text = "+" Then 'check which calculation needed
        CalValue = Val(Label3.Text) + Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
```

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    If Label4.Text = "-" Then 'check which calculation needed
        CalValue = Val(Label3.Text) - Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "x" Then 'check which calculation needed
        CalValue = Val(Label3.Text) * Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "÷" Then 'check which calculation needed
        'check if user trying to divide by 0
        If Val(TextBox1.Text) = 0 Then
            MsgBox("আহ! ০-দ্বারা ভাগ করা তো বৈধ নয়!") 'if so show a message and exit
            Exit Sub
        End If
        CalValue = Val(Label3.Text) / Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "%" Then 'check which calculation needed
        CalValue = (Val(Label3.Text) / 100) * Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = TextBox1.Text & " Gi " & Label1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
End Sub
```

প্রথমেই মনে করিয়ে দিচ্ছি, এ সাবরুটিন হলো সমান চিহ্নের বোতামের ক্লিক ইভেন্ট। প্রথমেই সবকিছু ঠিকমতো ডিসপ্লেতে ঢুকানো হয়েছে কি না তা দেখতে হবে। এ কাজের জন্য একটি ফাংশন বানিয়েছি- যার বর্ণনা শেষে আসবে। এ ফাংশনের কাজ হলো 'সত্য' কিংবা 'মিথ্যা' ফলাফল রিটার্ন করা। আমাদের উদ্দেশ্য সত্য ফলাফল পাওয়া। না পেলে সাবরুটিন থেকে বের হয়ে যেতে হবে। ফলাফল সত্য হলে পরের লাইনে যেয়ে একটি ডাবল টাইপ ভেরিয়েবল [CalValue] ঘোষণা করেছি যাতে থাকবে ফলাফল সংখ্যাটি। পরের লাইনে যেয়ে চেক করা হয়েছে কোন্ ক্যালকুলেন হবে [+ যোগ হবে কি?]। কথা সত্য হলে, এ সময় অবশ্যই লুকানো লেবেল৪-এ '+' লিখা থাকবে। ধরে নিন + লিখা আছে। এবার যোগ অঙ্ক কষতে হবে। পরের লাইনে আগে ঘোষিত ভেরিয়বলে সংখ্যা দু'টি যোগ করে সংরক্ষণ করা হয়েছে। এরপর লেবেল১-এ কিছু দৃশ্যমান তথ্য দেওয়া হয়েছে, ব্যবহারকারী কোন্ সংখ্যার সঙ্গে কোন্ সংখ্যা যোগ দিলেন তা বুঝানের জন্য। পরের লাইন দ্বারা যোগফলটি ৮টি ডেসিমেল প্লেইস পর্যন্ত প্রদর্শনের জন পরিবর্তন করে ডিসপ্লে ফিল্ডে দেখানো হয়েছে। পরের দু'টি লাইনের প্রথমটি দ্বারা লুকানো লেবেল-৩ এর মান মুছে ফেলার পর দ্বিতীয় লাইন দ্বারা লুকানো লেবেল২-এ প্রাপ্ত মানটি মেমোরি হিসাবে সংরক্ষিত হবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

সাবরুটিনের পরবর্তী ৪টি কোড ব্লকই মূলত একই ধরনের। এদের স্ট্রাকচার ও ফাংশন প্রায় একই। শুধুমাত্র –, x, ÷ ও % ক্যালকুলেশন লাইনগুলো একটু ভিন্ন- এই যা। সুতরাং সবকিছু পুনরায় বলার মানে নেই। তবে সামান্য কিছুটা পার্থক্য বিদ্যমান- যার কারণে কিছুটা ব্যাখ্যাও প্রয়োজন।

(ক) প্রথমত, ÷ অঙ্কের ক্ষেত্রে একটি ব্যাপার সকলেরই জানা- কোনো সংখ্যাকে ০ [শূন্য] দ্বারা ভাগ করা যায় না। এটা গাণিতিকভাবে অসম্ভব। সুতরাং কেউ যদি ০ দ্বারা ভাগ করার চেষ্টা করেন, তখন প্রোগ্রাম যাতে একটি মেসেজ প্রদর্শন করে সাবরুটিন থেকে বেরিয়ে পড়ে সে ব্যবস্থা করতে হবে। আমরা বাংলায় একটি মেসেজ প্রদর্শন করার ব্যবস্থা করেছি।

(খ) দ্বিতীয়ত, সবগুলো ক্যালকুলেশনের লাইনে ভ্যাল [VAL] নামক একটি ফাংশন ব্যবহার করেছি- লক্ষ করেছেন তো? এটার কাজ হলো ব্র্যাকেটের ভেতরস্থ স্ট্রিংটিকে সংখ্যায় পরিণত করা। এটা অতিরিক্ত সতর্কতা হেতু ব্যবহার করেছি। বাস্তবে অধিকাংশ ক্ষেত্রে ভিজুয়্যাল বেসিক এরূপ স্ট্রিংকে এমনিতেই সংখ্যায় পরিণত করে থাকে। তবে ক্যালকুলেটরের ক্ষেত্রে তো নিশ্চিতভাবে সংখ্যা বানানো ছাড়া বিকল্প নেই- তা-ই এ ব্যবস্থা।

ক্যালকুলেটরের কোডের ব্যাখ্যা প্রায় শেষ। দু'টি মাত্র কোড ব্লক বাকী আছে। এর প্রথমটি হলো ফাংশন ও দ্বিতীয়টি সাবরুটিন। প্রথমটি জরুরী আর দ্বিতীয়টি ব্যক্তিগত। চাইলে এটা দিতে পারেন- না দিলে অসুবিধে নেই। অবশ্য প্রোগ্রামার হিসাবে নাম ও তথ্যাদি তো থাকবে আপনারই, আমার নয়!

```
    'Function returns true if the conditions for calculation is fulfilled, otherwise returns false
Function CheckValidity()
    If TextBox1.Text <> "" And Label1.Text <> "" And Label3.Text <> "" And Label4.Text <> " Then" Then
        Return True
    End If
    Return False
End Function
'This sub is to show some info about the programmer- Button1 is 'About the Programmer' button at the top
 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    MsgBox("Engineer Azizul Bari" & vbCrLf & "Diploma in Science & Technology, England" & vbCrLf & _
    "Khanqa-E-Aminia-Asgaria Research Section" & vbCrLf & "Ali Centre, Subidbazar, Sylhet, Bangladesh" & _
      vbCrLf & "engineerazizulbari@gamil.com")
End Sub
```

ফাংশনের ফলাফল সত্যি হলে ক্যালকুলেশন হবে। এটা সত্যি হওয়ার শর্তগুলো হলো: ডিসপ্লে, প্রদর্শন লেবেল১, লুকানো লেবেল৩ এবং লুকানো লেবেল৪ খালি না হওয়া। এ সবগুলোর মধ্যে তথ্য থাকলেই বুঝা যাবে ক্যালকুলেশন হবে।

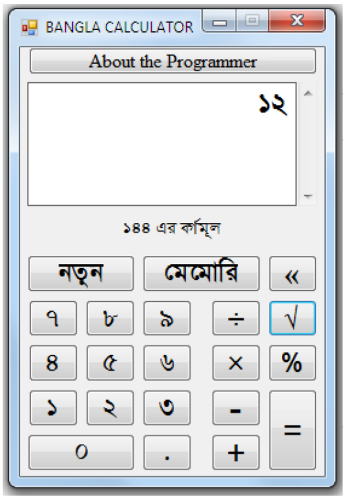
সবশেষে উপরোক্ত শেষের সাবরুটিনে দীর্ঘ লাইন কোড ফিল্ডে লিখার একটি উপায় দেখানো হয়েছে। সেসাথে একটি কোড দেওয়া হয়েছে যার কাজ ইনফিড বা রিটার্ন কিস্ট্রোকের ফাংশন আঞ্জাম দেওয়া। যথাক্রমে প্রথমটি হলো আন্ডারস্কোর [_] আর দ্বিতীয়টি vbCrLf কিওয়ার্ড।

বাকী থাকলো আপনার সৃষ্টি বাংলা ক্যালকুলেটর টেস্ট করে ডিস্ট্রিবিউট করা। পরের পৃষ্ঠায় এ সম্পর্কে কিছু তথ্য ও ব্যাখ্যা প্রদান করে এ টিউটোরিয়ালের ইতি টানবো।

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

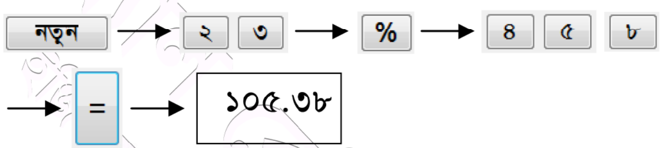
পরীক্ষণ ও বিস্তারণ: বাংলা ক্যালকুলেটর এ্যাপলিকেশনের প্রোগ্রামিং বা কোডিং পর্যায় শেষ হলো। এবার একে পরীক্ষা করতে হবে। এজন্য প্রয়োজন বিল্ড করে পরিচালনা। তাই এক্ষুণি পরীক্ষা শুরু হয়ে যাক। নিচের চিত্রে চলাকালীন জানালাটি দেখতে পাচ্ছেন।

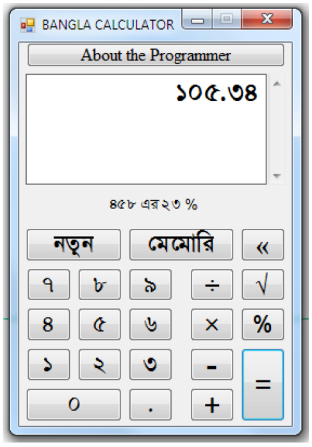


আমরা মূল ডিসপ্লে ফিল্ডে সংখ্যা ১২ দেখতে পাচ্ছি। ডিসপ্লের নিচে আছে '১৪৪ এর বর্গমূল' কথাটি। সুতরাং বুঝাই যাচ্ছে, ১৪৪ এর বর্গমূল হলো ১২- এই ক্যালকুলেন করেছি। এটা করতে প্রথমে ক্যালকুলেটরে আপনি যেভাবে সংখ্যা ঢুকাবেন তাহলো:



অনুরূপ ৪৫৮ -এর ২৩ শতাংশ বের করতে আপনি যে ভাবে সংখ্যা ঢুকাবেন তাহলো (ডানের চিত্র):





এ দু'টো ফাংশনের কাজ দেখানোর কারণ হলো, এগুলোই আমাদের বাংলা ক্যালকুলেটরের বিষেশ ফাংশন। বাকি চারটে পাটিগণিতের ফাংশন [+, - ÷ ও x] সচরাচর ব্যবহৃত ক্যালকুলেটরের মতোই কাজ করে। সুতরাং ওগুলোর অপারেশন এখানে দেখাচ্ছি না। তবে আপনার উচিৎ হবে সবগুলো ফাংশনই একে এক পরীক্ষা করা। আপনার এ সৃষ্টিকে তো শেষ পর্যন্ত গ্রাহকদের নিকট পৌঁছুনোর দরকার! আর এ কাজের নামই বিস্তারণ।

ভিজুয়্যাল স্টুডিও এক্সপ্রেস ফর ডেস্কটপ পূর্ণাঙ্গ এ্যাপলিকেশন বিস্তারণে আমাদেরকে বিষেশ সহযোগিতা করে। টেস্টিং স্তরের কাজ সম্পন্ন হওয়ার পর আপনি সহজেই কয়েকটির পদক্ষেপে সফটওয়্যার ডিস্ট্রিবিউটের জন্য ইনস্টলার তৈরি করতে পারেন। আমরা গ্রন্থের শেষে ইনস্টলার তৈরির পদ্ধতির ওপর বিস্তারিত আলোচনা করবো। এবার আসুন একটি উন্নতমানের বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার তৈরি করি! এ প্রোগ্রামটি দীর্ঘ, সময়সাক্ষেপক এবং কিছুটা জটিল। তবে প্রোগ্রাম করা শেষ হলে আপনি অবশ্যই আনন্দিত হবেন। কবি বলেন নি, এ সৃষ্টি সুখের উল্লাসে!

Application 3 : Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার।

সফটওয়্যার তৈরির লক্ষ্য: আমরা একটি বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার তৈরি করবো যাতেকরে ব্যবহারকারী অতি সহজে বিভিন্ন এককের মধ্যে সম্পর্ক, পরিবর্তন, সাংখ্যিক ফলাফল ইত্যাদি বের করতে সক্ষম হন।

ব্যাপ্তি: বিভিন্ন এককের মধ্যে শত শত সম্পর্ক বিদ্যমান। তবে আমাদের টার্গেট হলো সচরাচর ব্যবহৃত এককগুলো নিয়ে কাজ করা। সেমতে আমরা নিম্নের [পরের পৃষ্ঠার] টেবিলে দেওয়া ইউনিটগুলো নিয়ে কাজ করবো।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

দূরত্ব (Length)			দূরত্ব (Length)			দূরত্ব (Length)		
নাম	**সম্পর্ক**	**নাম**	**নাম**	**সম্পর্ক**	**নাম**	**নাম**	**সম্পর্ক**	**নাম**
মাইল	১=১.৬০৯৩	কি.মি	কি.মি	১=১০৯৩.৬১৩৩	গজ	ইঞ্চি	১=১/৬৩৩৬০	মাইল
মাইল	১=১৭৬০	গজ	কি.মি	১=১/১.৬০৯৩	মাইল	ইঞ্চি	১=১/৩৯৩৭০.০৭৮৮	কি.মি
মাইল	১=৫২৮০	ফুট	কি.মি	১=১০০০	মিটার	ইঞ্চি	১=১/৩৯.৩৭০০৭৮৮	মিটার
মাইল	১=১৬০৯.৩৪৪	মিটার	কি.মি	১=৩৯৩৭০.০৭৮৮	ইঞ্চি	ইঞ্চি	১=২৫.৪	মিমি
মাইল	১=১৬০৯৩৪৪	মিমি	কি.মি	১=৩২৮০.৮৩৯৯	ফুট	ইঞ্চি	১=১/৩৬	গজ
মাইল	১=৬৩৩৬০	ইঞ্চি	কি.মি	১=১০০০০০০	মিমি	ইঞ্চি	১=১/১২	ফুট
গজ	১=১/১৭৬০	মাইল	মিটার	১=১০০০	মিমি	মিমি	১=১/২৫.৪	ইঞ্চি
গজ	১=১/১০৯৩.৬১৩৩	কি.মি	মিটার	১=১/১০০০	কি.মি	মিমি	১=১/১০০০	মিটার
গজ	১=৩	ফুট	মিটার	১=৩.২৮০৮৪	ফুট	মিমি	১=১/৯১৪.৪	গজ
গজ	১=১/১.০৯৩৬	মিটার	মিটার	১=৩৯.৩৭০০৭৮৮	ইঞ্চি	মিমি	১=১/১৬০৯৩৪৪	মাইল
গজ	১=৯১৪.৪	মিমি	মিটার	১=১/১৬০৯.৩	মাইল	মিমি	১=১/১০০০০০০	কি.মি
গজ	১=৩৬	ইঞ্চি	মিটার	১=১.০৯৩৬	গজ	মিমি	১=১/৩০৪.৮	ফুট
ফুট	১=১/৫২৮০	মাইল	ফুট	১=১/৩.২৮০৮৪	মিটার	দূরত্ব একক রূপান্তরের জন্য মোট ৪২টি ফাংশন ব্যবহার করবো। ভাববেন না, বাস্তবে এ কাজ অতি সহজ।		
ফুট	১=১/৩২৮০.৮৩৯৯	কি.মি.	ফুট	১=৩০৪.৮	মিমি			
ফুট	১=১/৩	গজ	ফুট	১=১২	ইঞ্চি			

ক্ষেত্র (Area)		
নাম	সম্পর্ক	নাম
বর্গমাইল	১=২.৫৯	বর্গকি.মি
বর্গমাইল	১=১৮৬৮৩৪২৮.১১	বর্গমিটার
বর্গমাইল	১=২৭৮৭৮৪০০	বর্গফুট
বর্গমাইল	১=৩০৯৭৬০০	বর্গগজ
বর্গমাইল	১=৪০১৪৫০৫৬৫৮.০০৬৬	বর্গইঞ্চি
বর্গমাইল	১=২৫৮৯৯৮৮১১০৩৩৬	বর্গমিমি
বর্গকি.মি	১=১/২.৫৯	বর্গমাইল
বর্গকি.মি	১=১৫৫০০০৩১০০.০০৬২	বর্গইঞ্চি
বর্গকি.মি	১=১০০০০০০	বর্গমিটার
বর্গকি.মি	১=১১৯৫৯৯০.০৪৬৩	বর্গগজ
বর্গকি.মি	১=১০০০০০০০০০০০০	বর্গমিমি
বর্গকি.মি	১=১০৭৬৩৯১০.৪১৬৭	বর্গফুট
বর্গগজ	১=১/৩০৯৭৬০০	বর্গমাইল
বর্গগজ	১=১/১১৯৫৯৯০.০৪৬৩	বর্গকি.মি
বর্গগজ	১=৮৩৬১২৭.৩৬	বর্গমিমি
বর্গগজ	১=১/১.১৯৫৯৯	বর্গমিটার
বর্গগজ	১=১২৯৬	বর্গইঞ্চি
বর্গগজ	১=৯	বর্গফুট
বর্গমিমি	১=১/২৫৮৯৯৮৮১১০৩৩৬	বর্গমাইল
বর্গমিমি	১=১/৮৩৬১২৭.৩৬	বর্গগজ
বর্গমিমি	১=১/৬৪৫.১৬	বর্গইঞ্চি
বর্গমিমি	১=১/১০০০০০০	বর্গমিটার
বর্গমিমি	১=১/১০০০০০০০০০০০০	বর্গকি.মি
বর্গমিমি	১=১/৯২৯০৩.০৪	বর্গফুট

ক্ষেত্র (Area)		
নাম	সম্পর্ক	নাম
বর্গমিটার	১=১০০০০০০	বর্গমিমি
বর্গমিটার	১=১/১৮৬৮৩৪২৮.১১	বর্গমাইল
বর্গমিটার	১=১/১০০০০০০	বর্গকি.মি
বর্গমিটার	১=১০.৭৬৩৯১০৪১৬৭	বর্গফুট
বর্গমিটার	১=১.১৯৫৯৯	বর্গগজ
বর্গমিটার	১=১৫৫০.০০৩১	বর্গইঞ্চি
বর্গফুট	১=১/১০.৭৬৩৯১০৪১৬৭	বর্গমিটার
বর্গফুট	১=১/২৭৮৭৮৪০০	বর্গমাইল
বর্গফুট	১=১/৯	বর্গগজ
বর্গফুট	১=১৪৪	বর্গইঞ্চি
বর্গফুট	১=৯২৯০৩.০৪	বর্গমিমি
বর্গফুট	১=১/১০৭৬৩৯১০.৪১৬৭	বর্গকি.মি
বর্গইঞ্চি	১=১/৪০১৪৫০৫৬৫৮.০০৬৬	বর্গমাইল
বর্গইঞ্চি	১=১/১৫৫০০০৩১০০.০০৬২	বর্গকি.মি
বর্গইঞ্চি	১=১/১২৯৬	বর্গগজ
বর্গইঞ্চি	১=১/১৪৪	বর্গফুট
বর্গইঞ্চি	১=১/১৫৫০.০০৩১	বর্গমিটার
বর্গইঞ্চি	১=৬৪৫.১৬	বর্গমিমি

ওজন (Weight)		
কি.গ্রা	১=১/০.৯৩৩১০	সের
কি.গ্রা	১=১০০০	গ্রাম
কি.গ্রা.	১=১০০০/১১.৬৬	ভরি
গ্রাম	১=১/৯৩৩.১০	সের
গ্রাম	১=১/১০০০	কি.গ্রা
গ্রাম	১=১/১১.৬৬	ভরি
ভরি	১=১১.৬৬	গ্রাম

দূরত্বে আছে ৭টি, ক্ষেত্রতে আছে ৭টি, তাপে আছে ৩টি, ওজনে আছে ৪টি এবং সার্ভেতে আছে ৩টি একক। সুতরাং মোট ২৪টি একক নিয়ে আমাদের কাজ।

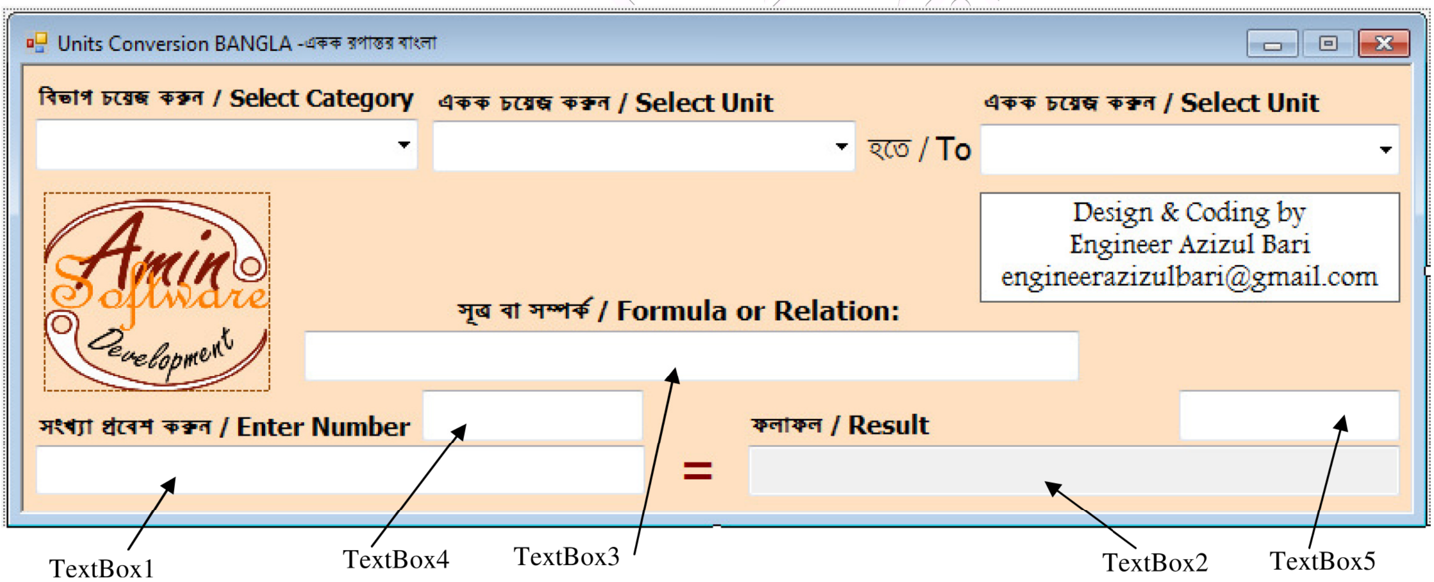
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

তাপ (Temperature)					
ডিগ্রী সেলসিয়াস	°F = (°C × 180/100) + 32)	ডিগ্রী ফারেনহাইট	ভরি	১=১/৮৫.৭৬৩২৯৩৩১	কি.গ্রা
ডিগ্রী ফারেনহাইট	°C = (°F - 32) × 100/180	ডিগ্রী সেলসিয়াস	ভরি	১=১/৮০.০১৭১৫	সের
কেলভিন	°C = (K - 273.15)	ডিগ্রী সেলসিয়াস	সের	১=০.৯৩৩	কি.গ্রা
ডিগ্রী সেলসিয়াস	K = (°C + 273.15)	কেলভিন	সের	১=৯৩৩.১	গ্রাম
কেলভিন	°F = (K × 9/5) – 459.67	ডিগ্রী ফারেনহাইট	সের	১=৮০.০১৭১৫	ভরি
ডিগ্রী ফারেনহাইট	K = (°F - 32) × 5 / 9 + 273.15	কেলভিন			
জমির হিসাব (Survey)					
বর্গফুট	১=১/৪৩৫৬০.১৭৪২	একর	একর	১=৪৩৫৬০.১৭৪২	বর্গফুট
বর্গফুট	১=১/৪৩৫.৬০১৭৪২	শতাংশ	শতাংশ	১=৪৩৫.৬০১৭৪২	বর্গফুট
একর	১=১০০	শতাংশ	শতাংশ	১=১/১০০	একর

উপরের টেবিলগুলোতে আমাদের ২৪টি এককের একটা আরেকটার সাথে সম্পর্ক কী তা দেখিয়েছি। এবার আমরা একক পরিবর্তন প্রোগ্রামের ইউজার ইন্টারফেইস ডিজাইনের কাজে লেগে যেতে পারি। আমাদের উদ্দেশ্য হলো ব্যবহারকারী প্রথমে চয়েজ করবেন ক্যাটাগরি। এরপর এ চয়েজকৃত ক্যাটাগরির কোন্ একক থেকে অপরটিতে পরিবর্তন করবের তা চয়েজ করবেন। সবশেষে চয়েজ করবেন কোন্ ইউনিটে পরিবর্তন হবে। অবশ্য এক পর্যায়ে যে একক থেকে পরিবর্তন করবেন তার মানও [সংখ্যা] তাকে একটি ফিল্ডে প্রবেশ করতে হবে।

অপর যে ব্যাপারটি তাহলো প্রোগ্রামটি নিরবচ্ছিন্নভাবে রান করবে। এর অর্থ হলো একক চয়েজ কিংবা মান ঢুকানোর পরই ফলাফল এসে যাবে। এজন্য আলাদাভাবে কোনো বোতামে ক্লিক করার দরকার পড়বে না। যাক ফর্মের একটি ডিজাইন নিচের চিত্রে তুলে ধরেছি। একটু দেখে নিন। এরপর বর্ণিত পদক্ষেপগুলো অনুসরণ করে নিজেই প্রজেক্ট তৈরি করে ফর্মটি ডিজাইন করুন।



১. একটি নতুন উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। নামকরণ করুন: Bangla - English Units Conversion।
২. ফর্মের সাইজ = ৪৮১, ৩০৬; ফর্মের Text = একক রূপান্তর বাংলা - ইংরেজি। বাংলা কথাগুলো ইউনিকোডে লিখবেন [বিজয় কিবোর্ডে ইউনিকোডে লিখার ফাংশন আছে]। ফর্মের ফন্ট = Tahoma, 12pt ; ফর্মের MaximizeBox = False; ফর্মের FormBorderStyle = Fixed3D; ফর্মের BackColor = 255,224,192। ব্যস! আর সকল প্রপার্টি অপরিবর্তিত থাকবে।
৩. ফর্মে যোগ করুন তিনটি কম্বোবক্স - ComboBox1, ComboBox2 এবং ComboBox3।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

৪. ফর্মে যোগ করুন ৭টি লেবেল ও ৫টি টেক্সট্ বাক্স। কোন্ লেবেলে কী লিখা থাকবে তা চিত্রে দেওয়া আছে। তবে ৫টি টেক্সট্ বাক্সের কোন্‌টি কোথায় হবে তা চিত্রে নির্দেশ মুতাবিক স্থাপন করুন। লেবেলের সকল লেখা মূলত তথ্য প্রদানের জন্য। আপনি লেবেলের Text প্রোপার্টিতে প্রথমে বাংলায় [ইউনিকোডে লিখবেন] এরপর ইংরেজিতে লিখবেন।
৫. টেক্সট্ বাক্স ১ ও ২ এর ফন্ট প্রেপার্টি = TonnyBanglaMJ, 14pt অথবা SutonnyMJ, 14pt। শুধুমাত্র টেক্সট্ বাক্স ২ এর ForeColor = Maroon এবং ReadOnly = True হবে। আমরা চাই এ বাক্সে ফলাফল প্রদর্শিত হবে। সুতরাং ব্যবহারকারী এতে যাতে কিছু লিখতে পারেন না- সেটা নিশ্চিত করবে শেষোক্ত প্রপার্টি।

ফর্মে আরো দু'টি বস্তু আছে: একটি টেক্সড্ বাক্স ও একটি পিকচার বক্স। এগুলো যুক্ত করা জরুরী নয়। কিসের জন্য এগুলো দিয়েছি তা-তো বুঝতেই পারছেন! তবে আপনি নিজেরটা দেবেন- এটা আশা করা যায়- কেমন?

যাক, এই তো! আমাদের প্রোগ্রামের ইউজার ইন্টারফেইস ডিজাইন হয়ে গেলো। একবার Start মেনু চয়েজ করে প্রোগ্রাম রান করে দেখুন। সব ঠিক আছে তো? আসুন তাহলে, মূল কাজে লেগে যাই- প্রোগ্রাম কোডিং!

কোডিং: ব্যক্তিগতভাবে আমি সহজ-সরল এ্যালগোরিথম ও লজিক ব্যবহারে বিশ্বাসী। এর কারণে যদি কোডিং দীর্ঘই হয় তাতে কোনো অসুবিধা নেই। প্রোগ্রাম বেঁটে করতে যেয়ে খুব জটিল ফাংশন, সাবরুটিন, ক্লাস, মডিউল ইত্যাদি তৈরি করার পক্ষপাতি আমি নই। হ্যাঁ, প্রয়োজনে এগুলোও করতে হবে- তবে সরলতা অনুসরণ করে। এভাবে প্রোগ্রাম করলে পরবর্তীতে ডিবাগিং অনেকটা সহজ হয়। যাক, আমাদের এ প্রোগ্রামটি বাস্তবে খুব একটা সহজ নয়। প্রথম কাজ হলো সঠিকভাবে বিভিন্ন কম্বো বক্স ও ফিল্ডে তথ্য প্রদর্শন করতে হবে। এজন্য প্রথমেই আমরা ডিফোল্ট জানালা প্রদর্শনের ব্যবস্থা করেছি। এরপর ব্যবহারকারীর চয়েজ অনুযায়ী প্রদর্শন ও ক্যালকুলেশন করতে হবে। তাহলে আসুন, কোডগুলো ধীরে ধীরে দেখে নিই ও প্রত্যেকটি সম্পর্কে ব্যাখ্যা তুলে ধরি।

আমরা তিনটি আলাদা এডিটর ফিল্ডে পূর্ণ কোড লিখেছি- আসলে লিখতে হয়েছে। এগুলো হলো:
১. ফর্মের কোড [Form1.vb*] ।
২. প্রদর্শনের জন্য একটি ফাংশন মডিউল [AllDisplayFunctionsModule.vb] ।
৩. সকল ক্যালকুলেশনের জন্য একটি ফাংশন মডিউল [CalculationModule.vb] ।

১. ফর্মের কোড [Form1.vb*]: এ কোডগুলোই প্রোগ্রাম চলাকালে সর্বপ্রথম রান করে। ফর্মের Public Class Form1 নামক সাবরুটিনের ভেতরই যাবতীয় কোড থাকতে হবে। এ সাবরুটিন থেকেই প্রয়োজনে আমরা বাকি দু'টো মডিউলের ফাংশনগুলো কল করতে হবে। সুতরাং ফর্মের ক্লাস সাবরুটিনটি কিভাবে কাজ করে তা জানা আমাদের জন্য একান্ত জরুরী। এর ভেতর আছে ১৮টি সাবরুটিন। এর মধ্যে ৭টি সিস্টেম ইভেন্ড সাবরুটিন এবং বাকি ১১টি আমাদের কর্তৃক ডিক্লায়রকৃত সাবরুটিন। অর্থাৎ এগুলো আমরা ইভেন্ট সাবরুটিনের ভেতর থেকে কল করে করে ব্যবহার করেছি বিভিন্ন কাজ সম্পাদন করতে। এ পর্যায়ে আমরা সবগুলো সাবরুটিনের নাম তালিকাভুক্ত করার প্রয়াস পাচ্ছি। নামের পাশে সাবরুটিনের কাজ কি তা-ও সংক্ষিপ্তভাবে বুঝিয়ে দিয়েছি।
১. সিস্টেম ইভেন্ট সাব: Form1_Load -এটা প্রোগ্রাম চলাকালে ফর্ম লোডের সময় রান করে।
২-৪. সিস্টেম ইভেন্ট সাব (২, ৩ ও ৪): ComboBox[1,2,3]_SelectedIndexChanged -এটা কম্বো বাক্সের ইনডেক্স সিলেকশনের ইভেন্ট সাব। আমরা এটা ধরে নিজেদের চাহিদানুযায়ী কাজ করেছি।
৫. সিস্টেম ইভেন্ট সাব: TextBox1_KeyPress -এটা হলো টেক্সড্ বাক্সে কিবোর্ড থেকে কিছু লিখার উভেন্ট সাব। আমার এটাকে ধরে শুধুমাত্র সংখ্যা ছাড়া যাতে আর কিছু ব্যবহারকারী কিবোর্ড থেকে প্রবেশ করতে পারেন না- সেটা নিশ্চিত করেছি। এভাবে নিয়ন্ত্রণ না করলে প্রোগ্রাম ক্রাশ করবে।
৬. সিস্টেম ইভেন্ট সাব: TextBox1_TextChanged -এটাও টেক্সট্ বাক্সের একটি ইভেন্ট যাকে আমরা ধরে কিছু ক্রিয়া সম্পাদন করেছি।

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

৭. আমাদের সৃষ্ট সাব: MileMileKilo() -এটা ফর্মের লোড [Form1_Load] সাব থেকে কল করা হয়েছে।
৮. আমাদের সৃষ্ট সাব: Combo1One() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
৯. আমাদের সৃষ্ট সাব: AreaMileKilo() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১০. আমাদের সৃষ্ট সাব: TempDegFahren() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১১. আমাদের সৃষ্ট সাব: WeightKiloSher() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১২. আমাদের সৃষ্ট সাব: SurveyFunction() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১৩. আমাদের সৃষ্ট সাব: CalculateAndDisplayLength() -এটার কাজ হলো 'দূরত্ব' ক্যাটাগরির কোন্ একক থেকে কোন্ একক পরিবর্তন হবে তা সনাক্ত করে মডিউলের ফাংশন কল করা। এরপর ফাংশন থেকে প্রাপ্ত ফলাফল ডিসপ্লে ফিল্ড টেক্সট বাক্স ২-এ প্রদর্শন করা। এ সাবটি উপরের (১৩ নং) সিস্টেম সাব থেকে কল করা হয়।
১৪. আমাদের সৃষ্ট সাব: CalculateAndDisplayArea() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'ক্ষেত্র' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৫. আমাদের সৃষ্ট সাব: CalculateAndDisplayTemp() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'তাপ' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৬. আমাদের সৃষ্ট সাব: CalculateAndDisplayWeight() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'ওজন' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৭. আমাদের সৃষ্ট সাব: CalculateAndDisplaySurvey() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'সার্ভে' ক্যাটাগরির সঙ্গে সম্পৃক্ত।

উপরোক্ত ১৭টি সাবরুটিনের সবই প্রোগ্রামের মেইন ক্লাস সাবরুটিন তথা Public Class Form1 -এর ভেতর অবস্থান করে। আগেই বলেছি দু'টি মডিউল আমরা নিজেই ডিফাইন করেছি আলাদাভাবে। এগুলোর ভেতর বেশ কিছু সাব ও ফাংশন আছে। মডিউল দু'টোর ব্যাপ্তি 'পাবলিক' হওয়ায় আমরা এসব সাব ও ফাংশন ফর্মের ক্লাস থেকে কল করার সুযোগ পেয়েছি। এ পর্যায়ে মডিউলে ডিফাইন করা প্রতিটি সাব সম্পর্কে বিস্তারিত বলার প্রয়োজন মনে করছি না। পরের পৃষ্ঠাগুলোতে আমরা সকল কোড একে একে তুলে ধরেছি। প্রথমে এসেছে Public Class Form1 -এর ভেতরস্থ সকল কোড। বিস্তারিত ব্যাখ্যা শেষে যথাক্রমে Public Module AllDisplayFunctionsModule -এর ভেতরস্থ কোড ও Public Module CalculationModule -এর ভেতরস্থ সকল কোড তুলে ধরেছি। প্রত্যেকটির ওপর বিস্তারিত ব্যাখ্যা তো অবশ্যই আছে। শিক্ষার্থীদের ধৈর্যধারণ ক্ষমতা আরো বেশি থাকা চাই। পুরো কোড তুলে ধরতে বইয়ের ১৫ পৃষ্ঠা লেগেছে। কোডের প্রতিটি লাইন ব্যাখ্যার আলোকে পাঠ করে বুঝতে হবে। সুতরাং আসুন, এবার পুরো কোড দেখে নিই ও বুঝার চেষ্টা করি। এই কোডটি কিভাবে কাজ করে তা যদি আমরা সঠিকভাবে বুঝতে সক্ষম হই তাহলে ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং সম্পর্কে তো বেশ দূরে অগ্রসর হয়ে গেলাম! তাই না? অবশ্য পরের প্রোগ্রামটি আরো একটু কঠিন! সুতরাং ধৈর্যের মাত্রা তখন আরো বাড়াতে হবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

১. ফর্মের কোড [Form1.vb*]

```
Public Class Form1
    'Global Variable Declarations
    'CatItem -Combobox1 Index, FromItem -Combobox2 Index, ToItem -Combobox3 Index
    Dim CatItem As Integer : Dim FromItem As Integer : Dim ToItem As Integer
    Dim ComBox2SelItemIdex As Integer 'to keep selected item from combo [combo2]
    Dim ComBox3SelItemIndex As Integer 'to keep selected item from combo [combo3]
    Dim EnteredNumber As Double 'to keep number item entered by the user
    Dim Combo1List(4) As String : Dim ComboInx0Combo2List(6) As String
    Dim ComboInx1Combo2List(6) As String
    Dim ComboboxInx2Combo2List(2) As String : Dim ComboboxInx3Combo2List(3) As String : Dim ComboboxInx4Combo2List(2)
As String
    '==================================================================
    'This Sub Initialises the default Text and Values. All changes during calculations.
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'Declare array variables for ComboBoxListItems and populate them.
        Combo1List = {"দূরত্ব / Length", "ক্ষেত্র / Area", "তাপ / Temperature", "ওজন / Weight", "জমির হিসাব / Survey"}
        ComboInx0Combo2List = {"মাইল / Mile (mi)", "কিলোমিটার / Kilometre (km)", "মিটার / Metre (m)", _
                    "মিলিমিটার / Millimetre (mm)", "গজ / Yard (yd)", "ফুট / Foot (ft)", "ইঞ্চি / Inch (in)"}
        ComboInx1Combo2List = {"বর্গমাইল / Sq. mi", "বর্গকিলোমিটার / Sq. Km", "বর্গমিটার / Sq. mt", "বর্গমিলিমিটার / Sq. mm", _
                    "বর্গগজ / Sq. yd", "বর্গফুট / Sq. ft", "বর্গইঞ্চি / Sq. in"}
        ComboboxInx2Combo2List = {"ডিগ্রী.সেলসিয়াস / °C", "ডিগ্রী.ফারেনহাইট / °F", "কেলভিন / K"}
        ComboboxInx3Combo2List = {"কিলোগ্রাম / kg", "সের Sher", "গ্রাম / g", "ভরি / Vori"}
        ComboboxInx4Combo2List = {"একর / Acre", "শতাংশ / Decimal", "বর্গফুট / Sq. ft"}
        MileMileKilo() 'local sub call only once during this load sub: initial display list in 3 comboBox
        ComboBox1.SelectedIndex = 0 'select 'length' catagory as default
        ComboBox2.SelectedIndex = 0 'select 'Mile' as Default
        ComboBox3.SelectedIndex = 1 'select 'kilometre' as dafault
        '======================================
□□□□Show default conversion value in textBox1□
        TextBox1.Text = "1"
        'Show default conversion result value in textBox2 -which is read only box
        TextBox2.Text = "1.6093"
        'Show default conversion realation in textBox3 -which is read only box
        TextBox3.Text = "১ মাইল = ১.৬০৯৩ কিলোমিটার"
        'Show default first value unit name in textBox4 & TextBox5, which are read only boxes
        TextBox4.Text = "মাইল" : TextBox5.Text = "কিলোমিটার"
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text) 'get the entred number into global variable
    End Sub
    'Any change will trigger calculation and display
    Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox1.SelectedIndexChanged
        CatItem = ComboBox1.SelectedIndex 'get the currently selected Index number into global variable
        'get selected items of ComboBox and Entered Number ino globals
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected index into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected index into global variable
        EnteredNumber = Val(TextBox1.Text)
        If CatItem = 0 Then Combo1One() 'sub call
        If CatItem = 1 Then AreaMileKilo() 'sub call
        If CatItem = 2 Then TempDegFahren() 'sub call
        If CatItem = 3 Then WeightKiloSher() 'sub call
        If CatItem = 4 Then SurveyFunction() 'sub call
    End Sub
    Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox2.SelectedIndexChanged
        'get selected items of ComboBox and Entered Number ino globals
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text) : FromItem = ComboBox2.SelectedIndex
        TextBox3.Text = UpdateRelationDisplay(CatItem, FromItem, ToItem) 'function retuns String
        TextBox4.Text = GetLeftWord(TextBox3.Text) 'function call to display unit in textBox3
        TextBox5.Text = GetRightWord(TextBox3.Text) 'function call to display TextBox5
        'Find what has been changed and take action accordingly
        If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
            CalculateAndDisplayLength() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
            CalculateAndDisplayArea() 'local sub call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" Then
            CalculateAndDisplayTemp() 'local sub call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" Then
            CalculateAndDisplayWeight() 'local sub call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
            CalculateAndDisplaySurvey() 'local sub call
        End If
    End Sub
    Private Sub ComboBox3_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox3.SelectedIndexChanged
        'get selected item into global
        ToItem = ComboBox3.SelectedIndex
        'get selected items of ComboBox and Entered Number ino globals
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text)
        TextBox3.Text = UpdateRelationDisplay(CatItem, FromItem, ToItem) 'function retuns String
        TextBox4.Text = GetLeftWord(TextBox3.Text) 'function call to display unit in textBox3
        TextBox5.Text = GetRightWord(TextBox3.Text) 'runction call to display TextBox5
        'Find what has been changed and take action accordingly
        If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
            CalculateAndDisplayLength() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
            CalculateAndDisplayArea() 'call local sub
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" Then
            CalculateAndDisplayTemp() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" Then
            CalculateAndDisplayWeight() 'call local sub
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
            CalculateAndDisplaySurvey() 'call local sub
        End If
    End Sub
    'This Sub makes sure only number [0-9 and decimal point once] entered in TextBox1
    Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
    End Sub
'This Sub makes calculation as soon as the value entered into text box1
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged
    'get selected items of ComboBox and Entered Number ino globals
    ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
    ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
    EnteredNumber = Val(TextBox1.Text)
    If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
        CalculateAndDisplayLength() 'call the 1st sub below
    End If
    If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
        CalculateAndDisplayArea() 'call the 2nd sub below
    End If
    If ComboBox1.SelectedItem = "তাপ / Temperature" Then
        CalculateAndDisplayTemp() 'call the 3rd sub below
    End If
    If ComboBox1.SelectedItem = "ওজন / Weight" Then
        CalculateAndDisplayWeight() 'call the 4th sub below
    End If
    If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
        CalculateAndDisplaySurvey() 'call the 5th Sub below
    End If
End Sub
'this sub adds catagory items in comboBox1
Sub MileMileKilo()
    Dim LoopContol As Integer
    ComboBox1.Items.Clear() 'must clear before adding list
    For LoopContol = 0 To 4
        ComboBox1.Items.Add(Combo1List(LoopContol))
    Next
End Sub
'this sub updates comboBox2 and ComboBox3 list Items, both box will have same list
'user chose length
Sub Combo1One()
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 6
        ComboBox2.Items.Add(ComboInx0Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
        ComboBox3.Items.Add(ComboInx0Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
    Next
    TextBox3.Text = "১ মাইল = ১.৬০৯৩ কিলোমিটার" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Area- this sub populates combo1 and 2
Sub AreaMileKilo()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both box will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 6
        ComboBox2.Items.Add(ComboInx1Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
        ComboBox3.Items.Add(ComboInx1Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
    Next
    TextBox3.Text = "১ বর্গমাইল = ১ বর্গমাইল" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Temperature- this sub populates combo1 and 2
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Sub TempDegFahren()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 2
        ComboBox2.Items.Add(ComboboxInx2Combo2List(LoopContol)) 'use declared array variable ComboInx2Combo2List
        ComboBox3.Items.Add(ComboboxInx2Combo2List(LoopContol)) 'use declared array variable ComboInx2Combo2List
    Next
    TextBox3.Text = "১ ডিগ্রী.সেলসিয়াস = ১ ডিগ্রী.সেলসিয়াস" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Weight- this sub populates combo1 and 2
Sub WeightKiloSher()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 3
        ComboBox2.Items.Add(ComboboxInx3Combo2List(LoopContol)) 'use declared array variable ComboInx3Combo2List
        ComboBox3.Items.Add(ComboboxInx3Combo2List(LoopContol)) 'use declared array variable ComboInx3Combo2List
    Next
    TextBox3.Text = "১ কিলোগ্রাম = ১ কিলোগ্রাম" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list

End Sub
'user chose Survey- this sub populates combo1 and 2
Sub SurveyFunction()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 2
        ComboBox2.Items.Add(ComboboxInx4Combo2List(LoopContol)) 'use declared array variable ComboInx4Combo2List
        ComboBox3.Items.Add(ComboboxInx4Combo2List(LoopContol)) 'use declared array variable ComboInx4Combo2List
    Next
    TextBox3.Text = "১ একর = ১ একর" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayLength()
    'Length category have 7 items
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then 'make sure there is a number in the entry text box
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মাইল / Mile (mi)" Then
            TextBox2.Text = CalculatedValueMileTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "কিলোমিটার / Kilometre (km)" Then
            TextBox2.Text = CalculatedValueKiloMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মিটার / Metre (m)" Then
            TextBox2.Text = CalculatedValueMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মিলিমিটার / Millimetre (mm)" Then
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            TextBox2.Text = CalculatedValueMilliMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "গজ / Yard (yd)" Then
            TextBox2.Text = CalculatedValueYardTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "ফুট / Foot (ft)" Then
            TextBox2.Text = CalculatedValueFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "ইঞ্চি / Inch (in)" Then
            TextBox2.Text = CalculatedValueInchTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayArea()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমাইল / Sq. mi" Then
            TextBox2.Text = CalculatedValueSqMileTo(EnteredNumber, ComBox3SelItemIndex) 'function call. Result displayed in textBox2
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গকিলোমিটার / Sq. Km" Then
            TextBox2.Text = CalculatedValueSqKilometreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমিটার / Sq. mt" Then
            TextBox2.Text = CalculatedValueSqMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমিলিমিটার / Sq. mm" Then
            TextBox2.Text = CalculatedValueSqMillimetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গগজ / Sq. yd" Then
            TextBox2.Text = CalculatedValueSqYardTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গফুট / Sq. ft" Then
            TextBox2.Text = CalculatedValueSqFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গইঞ্চি / Sq. in" Then
            TextBox2.Text = CalculatedValueSqinchTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayTemp()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "ডিগ্রী.সেলসিয়াস / °C" Then
            TextBox2.Text = CalculatedValueCelciusTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "ডিগ্রী.ফারেনহাইট / °F" Then
            TextBox2.Text = CalculatedValueFahrenheitTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "কেলভিন / K" Then
            TextBox2.Text = CalculatedValueKelvinTo(EnteredNumber, ComBox3SelItemIndex) 'function call
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayWeight()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "কিলোগ্রাম / kg" Then
            TextBox2.Text = CalculatedValueKgTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "সের Sher" Then '===
            TextBox2.Text = CalculatedValueSherTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "গ্রাম / g" Then
            TextBox2.Text = CalculatedValueGramTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "ভরি / Vori" Then
            TextBox2.Text = CalculatedValueVoriTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplaySurvey()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "একর / Acre" Then
            TextBox2.Text = CalculatedValueAcreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "শতাংশ / Decimal" Then
            TextBox2.Text = CalculatedValueDecimalTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "বর্গফুট / Sq. ft" Then
            TextBox2.Text = CalculatedValueSurveySqFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
    End If
End Sub
End Class
```

শেষ হলো ফর্মের ক্লাসের কোড। এবার আমরা লাইন থেকে লাইনের পূর্ণাঙ্গ ব্যাখ্যা তুলে ধরছি। শিক্ষার্থীরা যে অংশ বুঝে নিয়েছেন তা পাঠ না করে পরবর্তীতে যেতে পারেন। তবে না বুঝলে প্রয়োজনে বার বার পাঠ করে বুঝার চেষ্টা করবেন।

১. প্রথমেই আমরা বেশ ক'টি গ্লোবাল [সর্বত্র প্রাপ্য] ভেরিয়েবল ডিক্লায়ার করেছি। ক্লাসের প্রথমেই এগুলো ডিক্লায়ার করতে হয়। এদের মধ্যে আছে: (১) ৫টি ইন্টেজার টাইপ: (ক) CatItem -ক্যাটাগরি আইটেমের ইনডেক্স, (খ) FromItem -'হতে' আইটেমের ইনডেক্স, (গ) ToItem -'টু' আইটেমের ইনডেক্স, (ঘ) `ComBox2SelItemIdex` -ব্যবহারকারী কর্তৃক সিলেক্ট করা দ্বিতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা এবং `ComBox3SelItemIndex` -ব্যবহারকারী কর্তৃক সিলেক্ট করা তৃতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা। (২) ১টি ডবল টাইপ: `EnteredNumber` -ব্যবহারকারী কর্তৃক প্রবেশকৃত সংখ্যা। (৩) ৬টি স্ট্রিং টাইপ এ্যারে: (ক) Combo1List(4) -কম্বো বাক্স ১-এর ০-৪ এই ৫টি ক্যাটাগরির আইটেমের লিস্ট, (খ) ComboInx0Combo2List(6) - কম্বো বাক্স ২-এর ০-৬ এই ৭টি আইটেমের (০ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (গ) ComboInx1Combo2List(6) - কম্বো বাক্স ২-এর ০-৬ এই ৭টি আইটেমের (১ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (ঘ) ComboInx2Combo2List(2) - কম্বো বাক্স ২-এর ০-২ এই ৩টি আইটেমের (২ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (ঙ) ComboInx3Combo2List(3) - কম্বো বাক্স ২-এর ০-৩ এই ৪টি আইটেমের (৩ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট এবং

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

(চ) ComboInx4Combo2List(2) - কম্বো বাক্স ২-এর ০-২ এই ৩টি আইটেমের (৪ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট।

লক্ষ করুন, উক্ত ৬টি স্ট্রিং এ্যারের কাজ হলো ফর্ম লোড হওয়ার সময় ডিফোল্ট লিস্ট হিসাবে তিনটি কম্বো বাক্সে প্রদর্শন করা। সুতরাং আপনি প্রথমেই দেখতে পাবেন প্রথম কম্বো বাক্সে আছে চারটি ক্যাটাগরির লিস্ট- এর মধ্যে প্রথমটি হবে দূরত্ব। একই সময় কম্বো লিস্ট ২-এ থাকবে 'মাইল' ও কম্বো লিস্ট ৩-এ থাকবে 'কিলোমিটার'। তবে আপনি যে কোনো ক্যাটাগরি প্রথম কম্বোতে বদলালে অন্যগুলোও পরিবর্তন হবে সঠিকভাবে। ডিফোল্ট লিস্ট আসবে ফর্ম লোড ইভেন্ট থেকে আর পরিবর্তন হবে কম্বো বাক্সের সিলেক্টেডইনডেক্সচেঞ্জড্ [SelectedIndexChanged] সিস্টেম ইভেন্ট সাব থেকে। উক্ত সব ভেরিয়েবল বিভিন্ন সাবে ব্যবহৃত হবে- এজন্যই গ্লোবাল [সর্বত্র প্রাপ্য] হিসাবে ডিক্লায়ার করা হয়েছে। এবার প্রথম সাবের ব্যাখ্যা আসবে।

১. সিস্টেম ইভেন্ট সাব: Form1_Load -এর ব্যাখ্যা: এ সাবের উদ্দেশ্য হলো সর্বপ্রথম ফর্মে যা কিছু প্রদর্শিত হবে তার নির্দেশনা। প্রথমত, আমরা উপরে গ্লোবাল হিসেবে ডিক্লায়ার করা ৬টি এ্যারেতে ৬টি লিস্ট প্রবেশ করেছি। এরপর আমরা তিনটি কম্বো বাক্সের প্রথমটিতে ক্যাটাগরি, দ্বিতীয় ও তৃতীয়টিতে ডিফোল্ট একক লিস্ট প্রদর্শনের নির্দেশ দিয়েছে। এটা করতে যেয়ে ব্যবহার করেছি ফর্মের ক্লাস ইভেন্ট সাবের অন্যত্র নিজেদের সৃষ্ট একটি সাব ব্যবহার করে, যার নাম MileMileKilo()। সুতরাং এ পর্যন্ত কোড রান করার পর তিনটি কম্বো বাক্সে তিনটি লিস্ট লিখা হবে। প্রথম বাক্সে সর্বদা একই লিস্ট থাকবে- কারণ এটা হলো ক্যাটাগরি লিস্ট। এর ডিফোল্ট [দৃশ্যমান লেখা] আইটেম হবে 'দূরত্ব / Length'। সুতরাং দ্বিতীয় ও তৃতীয়টিতে ডিফোল্ট হিসাবে থাকবে মাইল-কিলোমিটার। লক্ষ করুন, - কোন্ ইনডেক্টকৃত লেখা কম্বো বাক্সে থাকবে তা নির্ধারণ করতে নির্দেশ দিয়েছি এই কোডগুলো দ্বারা: `ComboBox1.SelectedIndex = 0, ComboBox2.SelectedIndex = 0` এবং `ComboBox3.SelectedIndex = 1`। অর্থাৎ ডিফোল্ট হিসেবে প্রদর্শিত হবে যথাক্রমে: দূরত্ব / Length, মাইল / Mile (mi) এবং কিলোমিটার / Kilometer (km)। টেক্সট্ বাক্স ১, ২, ৩, ৪ ও ৫-এ ডিফোল্ট মানের সাথে সামঞ্জস্য রেখে কিছু তথ্য ব্যবহারকারীকে দেওয়া হয়েছে। সাবের শেষের তিনটি লাইনের কাজ হলো কম্বো বাক্স ২ ও ৩ এর ইনডেক্স সংখ্যা এবং প্রবেশ ফিল্ডের সংখ্যাটি ডিক্লায়ার করা গ্লোবাল ভেরিয়েবলে ঢুকানো। উল্লেখ্য এই ঢুকানোর কাজ হবে যতোবারই ইনডেক্স ও সংখ্যা পরিবর্তন হবে। শেষ হলো Form1_Load -সাব এর ব্যাখ্যা।

২-৪. সিস্টেম ইভেন্ট সাব (২, ৩ ও ৪): -এর ব্যাখ্যা: এগুলো তিনটি কম্বো বাক্সের SelectedIndexChanged -সিস্টেম ইভেন্ট সাব। ব্যবহারকারী এ তিনটি বাক্সের আইটেম মুক্তভাবে চয়েজ করবেন। আমাদের কাজ হলো কোন্‌টি চয়েজ করেছেন তা জানা ও সে মুতাবিক ক্রিয়া সম্পাদন করা।

সাব ২-এর প্রথম ৩ লাইন দ্বারা CatItem, `ComBox2SelItemIdex`, এবং `ComBox3SelItemIndex` -এ তিনটি গ্লোবাল হিসেবে ডিক্লায়ার করা ভেরিয়েবলে মান ঢুকিয়েছি। ফলে এগুলো যে কোনো স্থান ও সাব থেকে কাজে লাগানো যাবে। সাবের CatItem ভেরিয়েবল ইনডেক্ট [0-8] বলে দেবে কোন্ ক্যাটাগরি চয়েজ করা হয়েছে। যে'টি চয়েজ হয়েছে সে মুতাবিক যথাক্রমে ৫টি সাবরুটিনের [Combo1One(), AreaMileKilo(), TempDegFahren(), WeightKiloSher(), SurveyFunction()] সঠিকটি কল করা হয়েছে। এ প্রতিটি সাবের কাজ হলো কম্বো বাক্সে সঠিক লিস্ট লেখা ও ডিফোল্ট একটি আইটেম দেখানো। আমরা এ সাবগুলোর ব্যাখ্যার সময় আরো বিস্তারিত বলবো। বাকী দু'টো ভেরিয়েবল ক্যালকুলেশনের জন্য ব্যবহৃত হয়েছে।

সাব ৩-এর প্রথম তিন লাইনে উপরে বর্ণিত ভেরিয়েবলগুলো পুনরায় আপডেট করা হয়েছে- কারণ নতুনভাবে এক বা একাধিক সংখ্যা ও মান পরিবর্তন হয়েছে। চতুর্থ লাইন দ্বারা `FromItem`-নামক গ্লোবাল ইন্টেজার ভেরিয়েবলে দ্বিতীয় কম্বো বাক্সের ইনডেক্স মান ঢুকানো হয়েছে। পরের তিনটি লাইনের কাজ হলো যথাক্রমে টেক্সট্ বাক্স ৩, ৪ ও ৫-এ সম্পর্ক, কোনটি থেকে একক পরিবর্তন হবে এবং কোন্ অপর এককে পরিবর্তন হওয়া উদ্দেশ্য তা প্রদর্শন করা।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

অবশ্য এ কাজের জন্য `AllDisplayFunctionsModule.vb` -এ ডিফাইন করা তিনটি ফাংশন তথা `UpdateRelationDisplay`, `GetLeftWord` এবং `GetRightWord`-কে কল করা হয়েছে। পরের চারটি -ব্লকে আমরা কম্বো বাক্স-১-এ কোন্ ক্যাটাগরি চয়েজ করা আছে তা পরীক্ষা করে যথাযথ তথ্য প্রদর্শনের জন্য নিচে ডিফাইন করা সাবরুটিন কল করেছি। এসব সাবরুটিন সম্পর্কে ব্যাখ্যা পরে আসবে।
সাব ৪-এর কাজ উপরের সাবের মতোই। সুতরাং এ সম্পর্কে অতিরিক্ত কিছু বলার নেই- শুধুমাত্র এটুকু যে, এটা হলো কম্বো বাক্স ৩-এর `SelectedIndexChanged` -ইভেন্ট সাব। সুতরাং সে মুতাবিক সবকিছু আবার আপডেটও করতে হবে।

৫. সিসেটম ইভেন্ট সাব: TextBox1_KeyPress -এর ব্যাখ্যা: টেক্সড্ বাক্স ১-এ ব্যবহারকারী সংখ্যা প্রবেশ করবেন। কিন্তু সংখ্যা ছাড়া অন্য কোনো অক্ষর বা ক্যারাক্টারও ঢুকাতে পারেন। ফলে অঙ্ক কষার নির্দেশনার সময় প্রোগ্রাম অভিযোগ তুলবে- সংখ্যা ছাড়া অন্য কিছু কেনো ঢুকানো হলো? এ সাব নিশ্চিত করে, যাতেকরে প্রোগ্রাম কোনো অভিযোগ তুলতে না পারে। শুধুমাত্র আস্ত ও দশমিক সংখ্যা ছাড়া অন্য কোনো কিছুই এ টেক্টড্ বাক্সে যাতে ঢুকানো যায় না, সেটা নিশ্চিত করতে যেয়ে এই কোড।

৬. সিস্টেম ইভেন্ট সাব: TextBox1_TextChanged -এর ব্যাখ্যা: টেক্সট্ বাক্সের এই ইভেন্ট ধরে আমরা ক্যালকুলেশন করে ফলাফল প্রদর্শন করেছি। অর্থাৎ সংখ্যা ঢুকানোর পরই এই সাব সক্রিয় হয়ে ওঠে। প্রথমেই সংগ্রহ করেছি তিনটি গুরুত্বপূর্ণ গ্লোবাল মান, যথা: `ComBox2SelItemIdex`, `ComBox3SelItemIdex` এবং `EnteredNumber`। এগুলো দ্বারাই ক্যালকুলেশন করবো। এরপর আমরা পাঁচটি If .... End IF টেস্ট দ্বারা সঠিকটি নির্ধারণ করে নিচের ৫টি [CalculateAndDisplayLength(), CalculateAndDisplayArea(), CalculateAndDisplayTemp(), CalculateAndDisplayWeight(), CalculateAndDisplaySurvey()] সাবরুটিনের একটি কল করেছি ক্যালকুলেশন করে ফলাফল প্রদর্শনের জন্য। লক্ষ করুন, ৫টি টেস্টের উদ্দেশ্য হলো কোন্ ক্যাটাগরি তা জানা ও সে মুতাবিক সাবরুটিন কল করা। ঐ ৫টি সাবরুটিনের ব্যাখ্যাও একে একে এখন তুলে ধরছি।

৭. আমাদের সৃষ্ট সাব: MileMileKilo() -এর ব্যাখ্যা: আগেই বলেছি ফর্মের লোড ইভেন্ট সাব [১ম সাব] থেকে এটা কল করা হয়েছে। এটার কাজ হলো কম্বো বাক্স-১ এর লিস্ট তৈরি করা। লক্ষ করুন, লোড সাবে ডিফাইন করা Combo1List নামক এ্যারে ভেরিয়েবলটি ব্যবহার করেছি লুপে। ইনডেক্স ০-৪ দ্বারা লিস্ট বানিয়েছি ঐ ভেরিয়েবলে সংরক্ষিত আইটেম নাম দ্বারা। সাবে মোট ৫টি মাত্র লাইন আছে। প্রথম লাইনে `LoopContol` নামক একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি। দ্বিতীয় লাইন দ্বারা কম্বো বাক্সের আগের আইটেম লিস্ট মুছে দেওয়া হয়েছে। এরপর ফর-নেক্সট্ লুপের মাধ্যমে পুরো লিস্টটি কম্বো বাক্সে যুক্ত করেছি।

৮. আমাদের সৃষ্ট সাব: Combo1One() -এর ব্যাখ্যা: এটাও উপরোক্ত (৭ম) সাবের মতো কাজ করে। অবশ্য এক্ষেত্রে পার্থক্য হলো কম্বো বাক্স ২ ও ৩-এর লিস্ট বানানো। এখানে বলা হচ্ছে, যেহেতু ব্যবহারকারী একটি ক্যাটাগরি চয়েজ করেছেন তাই সে মুতাবিক বাকী দু'টো বাক্সের লিস্টে কি হবে তাও আপডেট করা জরুরী। এ কাজটিই হলো এ সাবের উদ্দেশ্য। লক্ষ করুন, উভয় কম্বো বাক্সের লিস্ট ফর-নেস্ট লুপের মাধ্যমে বানানোর পর যে দু'টো সিলেক্ট করে দেখানো আমাদের উদ্দেশ্য তা শেষের লাইন দু'টোতে নির্দেশ দিয়েছি। একই সময় টেক্সট্ বাক্স-৩-এ সম্পর্ক দেখিয়েছি- যা এ ক্ষেত্রে হলো: ১ মাইল = ১.৬০৯৩ কিলোমিটার।

৯. আমাদের সৃষ্ট সাব: AreaMileKilo() -এর ব্যাখ্যা: এটার কাজও উপরোক্ত সাবের মতো। তবে এখানে বলা হচ্ছে ব্যবহারকারী 'ক্ষেত্র / Area' ক্যাটাগরি চয়েজ করেছেন, তাই সে মুতাবিক বাকী দুটো বাক্সের লিস্ট তৈরি করা দরকার।

১০. আমাদের সৃষ্ট সাব: TempDegFahren() -এর ব্যাখ্যা: এ সাবের কাজও অনুরূপ। এ ক্ষেত্রে যেহেতু ব্যবহারকারী 'তাপ / Temperature' চয়েজ করেছেন তাই সে মুতাবিক বকী দু'টো বাক্সের লিস্ট তৈরি করতে বলা হয়েছে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

১১. আমাদের সৃষ্ট সাব: WeightKiloSher() -এর ব্যাখ্যা: এটাও একইভাবে কাজ করে। এখানে যেহেতু চয়েজ করা হয়েছে 'ওজন / Weight' তাই সে মুতাবিক বাকী দু'টো বাক্সের লিস্ট তৈরির নির্দেশ দেওয়া হয়েছে।

১২. আমাদের সৃষ্ট সাব: SurveyFunction() -এর ব্যাখ্যা: এটার কাজ হলো 'জমির হিসাব / Survey' মুতাবিক বাকী দু'টো বাক্সের লিস্ট বানানো।

১৩. আমাদের সৃষ্ট সাব: CalculateAndDisplayLength() -এর ব্যাখ্যা: এ সাবও পরবর্তী আরো ৪টি সাবের ক্রিয়া অনুরূপ। সুতরাং প্রথম এ সাব সম্পর্কে বিস্তারিত ব্যাখ্যা তুলে ধরাই যথেষ্ট মনে করছি। এ সাবটি ডাকা হয় দ্বিতীয় ও তৃতীয় কম্বো বাক্সের সিলেকশন পরিবর্তন [selectedIndexChanged] ইভেন্ট সাব থেকে। এছাড়া এটি উপরোক্ত [৬ নং] সিস্টেম সাব থেকেও কল করা হয়। আমাদের উদ্দেশ্য হলো ব্যবহারকারী যে হিসাব চান তা যেনো তাড়াতাড়ি পেয়ে যান। সুতরাং আমরা লিস্ট পরিবর্তন ও সংখ্যা পরিবর্তন- এ উভয় সিস্টেম ইভেন্টকে ধরে ক্যালকুলেশন সেরে ফলাফল প্রদর্শনের ব্যবস্থা করেছি।
লক্ষ করুন, প্রতিটি সাব শুরু হয়েছে একটি If .... End If টেস্ট দ্বারা। এর ভেতর আবার আরো কিছু If .... End If টেস্ট আছে। বাইরের টেস্ট দ্বারা নিশ্চিত করা হচ্ছে যে, সংখ্যা টেক্স বাক্স খালি নয় এবং এর মধ্যে প্রবেশকৃত সংখ্যাটি গ্রহণযোগ্য। এরপর ভেতরের টেস্ট দ্বারা নির্ধারিত হয়েছে ক্যাটাগরি 'দূরত্ব / Length' এবং দ্বিতীয় কম্বো বাক্সে "মাইল / Mile (mi)" ইউনিট সিলেক্ট করা আছে কি না। যদি উভয় কন্ডিশন সত্য হয় তাহলে আমরা কল করেছি 'CalculationModule.vb'-এ ডিফাইন করা 'CalculatedValueMileTo()' -নামক ফাংশনটি। এ ফাংশন সম্পর্কে পরে বলবো। এখানে এটুকু বলে রাখছি, এটির কাজ হলো দূরত্ব ক্যাগরির দ্বিতীয় কম্বোতে 'মাইল' এর মাত্রা [সংখ্যা] কোন্ দূরত্ব ইউনিটে পরিবর্তন হবে তা নির্ধারণ করে ক্যালকুলেন করা ও ফলাফল ফেরত পাঠানো। লক্ষ করুন: TextBox2.Text = CalculatedValueMileTo() -এ লাইনটি একই সাথে দু'টি কাজ করছে: (ক) ফাংশন কল করছে ও ফেরত প্রাপ্ত সংখ্যাটি ডিসপ্লে ফিল্ডে [অর্থাৎ টেক্সড্ বাক্স-২ -এ] প্রদর্শনও করে দিচ্ছে। ঠিক অনুরূপ সাবরুটিনের বাকী ৬টি টেস্টও মাইল থেকে কোন্ ইউনিটে পরিবর্তিত হবে সেটা নিশ্চিত করে একই ফাংশন কল করে রেজাল্ট প্রদর্শনের কাজ করছে। মনে রাখবেন ফাংশন কল থেকে একটি মাত্র ভ্যালু রিটার্ন হবে- এবং তা নির্ধারিত হবে মাইল থেকে কোন্ ইউনিটে পরিবর্তন হবে সে মুতাবিক। দূরত্বের সাথে মোট ৭টি ইউনিট জড়িত থাকায় ৭টি আলাদা ফাংশন তৈরি করতে হয়েছে। অনুরূপ অন্যান্য ক্ষেত্রেও ক'টি ইউনিটের সঙ্গে প্রত্যেকটি ক্যাটাগরি জড়িত সে হিসাবে ফাংশনও বানিয়েছি।

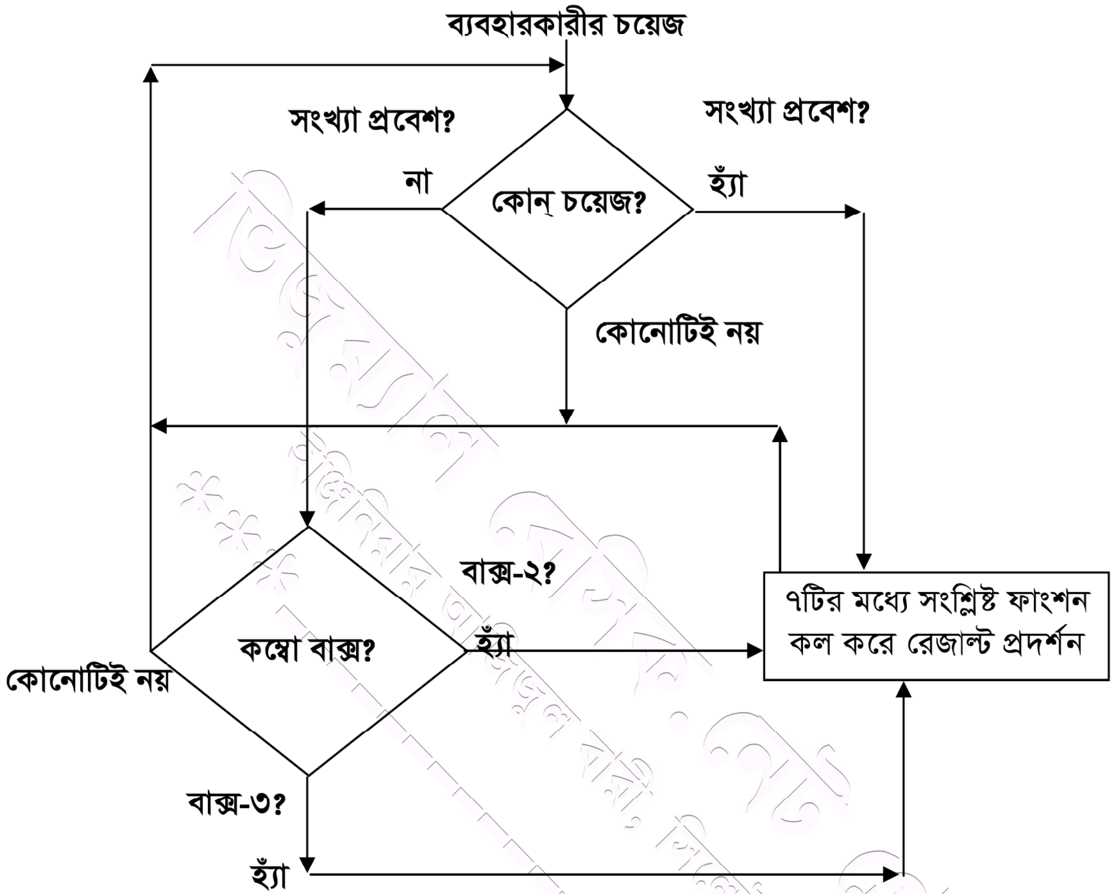
উপরের ব্যাখ্যা থেকে আমরা বুঝতে পারলাম যে, আমাদের সৃষ্ট ৫টি সাব [১৩ থেকে ১৭] মূলত টেক্সট্ বাক্সে সংখ্যা প্রবেশ কিংবা উভয় কম্বো বাক্সে [১ ও ২ -এর যে কোনোটিতে] আইটেম সিলেক্ট করার পরই কাজ শুরু করে। কোনটি সক্রিয়া হবে তা নির্ভর করে কোন্ আইটেম ব্যবহারকারী সিলেক্ট করছেন বা [সংখ্যা প্রবেশ করার ক্ষেত্রে] কোনটি সিলেক্ট করেছিলেন।

শিক্ষার্থীদের সুবিধার্থে পরের পৃষ্ঠায় উপরোক্ত ৫টি ফাংশনের ক্রিয়ার একটি ফ্লো চার্ট [পরম্পরা ক্রিয়াচিত্র] উপস্থাপন করেছি। এটা ভালো করে স্টাডি করে নিলে বিষয়টি সহজেই বুঝে এসে যাবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

বর্ণিত ৭টি সাবরুটিনের পরম্পরা ক্রিয়াচিত্র [ফ্লো চার্ট]



উপরের পরম্পরা ক্রিয়াচিত্র আমাদের প্রোগ্রামের এ্যালগোরিথম ও লজিক স্পষ্ট করেছে। ফর্ম প্রদর্শের পর আমরা অপেক্ষায় থাকি ব্যবহারকারীর পক্ষ থেকে ইনপুটের। তিনি হয় কোনো কম্বো বাক্সের লিস্ট বদলাবেন তার চাহিদা মতো না হয় টেক্সট্ ফিল্ড-১ -এ কোনো সংখ্যা লিখবেন। উভয় ক্রিয়া সম্পাদনকে আমরা কাজে লাগিয়েছি। আপনি যদি উপর থেকে এ্যারোগুলো অনুসরণ করেন তাহলে লজিকটি বুঝতে পারবেন। তিনটি ক্রিয়ার যে কোনো একটি ব্যবহারকারীকে করতেই হবে: যথা- ১. কোনো ক্রিয়া করবেন না। ২. প্রথমে কম্বো বাক্সের লিস্ট বদলিয়ে নিজের কাঙ্ক্ষিত একক থেকে একক পরিবর্তনের জন্য আইটেম চয়েজ করবেন এবং ৩. শেষে অথবা প্রথমেই ফিল্ডে সংখ্যাটি লিখবেন। প্রোগ্রাম বন্ধ করা ছাড়া তার পক্ষ থেকে আর কোনো ক্রিয়া সম্পাদনের পথ খোলা নেই।

এবার আমরা একটি কল্পিত অবস্থা নিয়ে ভাবতে পারি। যেমন: ১. প্রোগ্রাম চালু হাওয়ার পর ব্যবহারকারী ‘ওজন / Weight’ ক্যাটাগরি চয়েজ করলেন। এরপর দ্বিতীয় কম্বোতে যেয়ে চয়েজ করলেন ‘ভরি’ ও তৃতীয়টিতে যেয়ে চয়েজ করলেন ‘গ্রাম’। বুঝাই যাচ্ছে তিনি ভরি থেকে গ্রামের হিসাবটা জানতে চান। সবশেষে তিনি প্রবেশ করলেন কতো ভরি-তথা একটি সংখ্যা। উপরের ফ্লো চার্টে আমাদের প্রথম প্রশ্নের জবাব হবে ‘না’- এর পরের প্রশ্নের জবাব আসবে ‘হ্যাঁ’। এরপর চতুর্ভুজের ভেতরের ক্রিয়াগুলো সম্পাদিত হবে। যাক, আশাকরি পাঠকরা পরম্পরা ক্রিয়াচিত্র থেকে কিছুটা উপকৃত হয়েছেন। তবে প্রোগ্রামের অনেক কোড সম্পর্কে এখনো আমাদের জানার আছে। তাহলে আসুন, ওগুলোর ওপর বিস্তারিত বিবরণে যাই।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

২. প্রদর্শনের জন্য ফাংশন মডিউল [AllDisplayFunctionsModule.vb] -এর কোড

```
Public Module AllDisplayFunctionsModule
    'Declaration of arrays 'block1 - length to length
    Dim CatLenMileTo() As String = {"১ মাইল = ১ মাইল", "১ মাইল = ১.৬০৯৩ কিলোমিটার", "১ মাইল = ১৬০৯.৩৪৪ মিটার", "১ মাইল = ১৬০৯৩৪৪ মিলিমিটার", _
    "১ মাইল = ১৭৬০ গজ", "১ মাইল = ৫২৮০ ফুট", "১ মাইল = ৬৩৩৬০ ইঞ্চি"}
    Dim CatLenKiloTo() As String = {"১ কিলোমিটার = ১/১.৬০৯৩ মাইল", "১ কিলোমিটার = ১ কিলোমিটার", "১ কিলোমিটার = ১০০০ মিটার", "১ কিলোমিটার = ১০০০০০০ মিলিমিটার", _
   "১ কিলোমিটার = ১০৯৩.৬১৩৩ গজ", "১ কিলোমিটার = ৩২৮০.৮৩৯৯ ফুট", "১ কিলোমিটার = ৩৯৩৭০.০৭৮৮ ইঞ্চি"}
    Dim CatLenMeterTo() As String = {"১ মিটার = ১/১৬০৯.৩ মাইল", "১ মিটার = ১/১০০০ কিলোমিটার", "১ মিটার = ১ মিটার", "১ মিটার = ১০০০ মিলিমিটার", _
    "১ মিটার = ১.০৯৩৬ গজ", "১ মিটার = ৩.২৮০৮৪ ফুট", "১ মিটার = ৩৯.৩৭০০৭৮৮ ইঞ্চি"}
    Dim CatLenMilliMeterTo() As String = {"১ মিলিমিটার = ১/১৬০৯৩৪৪ মাইল", "১ মিলিমিটার = ১/১০০০০০০ কিলোমিটার", "১ মিলিমিটার = ১/১০০০ মিটার", "১ মিলিমিটার = ১ মিলিমিটার", _
    "১ মিলিমিটার  = ১/৯১৪.৪ গজ", "১ মিলিমিটার = ১/৩০৪.৮ ফুট", "১ মিলিমিটার = ১/২৫.৪ ইঞ্চি"}
    Dim CatLenYardTo() As String = {"১ গজ = ১/১৭৬০ মাইল", "১ গজ = ১/১০৯৩.৬১৩৩ কিলোমিটার", "১ গজ = ১/১.০৯৩৬ মিটার", "১ গজ = ৯১৪.৪ মিলিমিটার", _
    "১ গজ = ১ গজ", "১ গজ = ৩ ফুট", "১ গজ = ৩৬ ইঞ্চি"}
    Dim CatLenFootTo() As String = {"১ ফুট = ১/৫২৮০ মাইল", "১ ফুট = ১/৩২৮০.৮৩৯৯ কিলোমিটার", "১ ফুট = ১/৩.২৮০৮৪ মিটার", "১ ফুট = ৩০৪.৮ মিলিমিটার", _
    "১ ফুট = ১/৩ গজ", "১ ফুট = ১ ফুট", "১ ফুট = ১২ ইঞ্চি"}
    Dim CatLenInchTo() As String = {"১ ইঞ্চি  = ১/৬৩৩৬০ মাইল", "১ ইঞ্চি = ১/৩৯৩৭০.০৭৮৮ কিলোমিটার", "১ ইঞ্চি = ১/৩৯.৩৭০০৭৮৮৪ মিটার", "১ ইঞ্চি = ২৫.৪ মিলিমিটার", _
    "১ ইঞ্চি = ১/৩৬ গজ", "১ ইঞ্চি = ১/১২ ফুট", "১ ইঞ্চি = ১ ইঞ্চি"}
    '++++++++++++++ 'Block2 - Area to Area
    Dim CatAreaSqMileTo() As String = {"১ বর্গমাইল = ১ বর্গমাইল", "১ বর্গমাইল = ২.৫৯ বর্গকিলোমিটার", "১ বর্গমাইল = ১৮৬৮৩৪২৮.১১ বর্গমিটার", "১ বর্গমাইল = ২৫৮৯৯৮৮১১০৩৩৬ বর্গমিলিমিটার", _
   "১ বর্গমাইল = ৩০৯৭৬০০ বর্গগজ", "১ বর্গমাইল = ২৭৮৭৮৪০০ বর্গফুট", "১ বর্গমাইল = ৪০১৪৫০৫৬৫৮.০০৬৬ বর্গইঞ্চি"}
    Dim CatAreaSqKiloTo() As String = {"১ বর্গকিলোমিটার = ১/২.৫৯ বর্গমাইল", "১ বর্গকিলোমিটার = ১ বর্গকিলোমিটার", "১ বর্গকিলোমিটার = ১০০০০০০ বর্গমিটার", "১ বর্গকিলোমিটার = ১০০০০০০০০০০০০ বর্গমিলিমিটার", _
   "১ বর্গকিলোমিটার = ১১৯৫৯৯০.০৪৬৩ বর্গগজ", "১ বর্গকিলোমিটার = ১০৭৬৩৯১০.৪১৬৭ বর্গফুট", "১ বর্গকিলোমিটার = ১৫৫০০০৩১০০.০০৬২ বর্গইঞ্চি"}
    Dim CatAreaSqMeterTo() As String = {"১ বর্গমিটার = ১/১৮৬৮৩৪২৮.১১ বর্গমাইল", "১ বর্গমিটার = ১/১০০০০০০ বর্গকিলোমিটার", "১ বর্গমিটার = ১ বর্গমিটার", "১ বর্গমিটার = ১০০০০০০ বর্গমিলিমিটার", _
    "১ বর্গমিটার = ১.১৯৫৯৯ বর্গগজ", "১ বর্গমিটার = ১০.৭৬৩৯১০ বর্গফুট", "১ বর্গমিটার = ১৫৫০.০০৩১ বর্গইঞ্চি"}
    Dim CatAreaSqMilliMeterTo() As String = {"১ বর্গমিলিমিটার = ১/২৫৮৯৯৮৮১১০৩৩৬ বর্গমাইল", "১ বর্গমিলিমিটার = ১/১০০০০০০০০০০০০ বর্গকিলোমিটার", "১ বর্গমিলিমিটার = ১/১০০০০০০ বর্গমিটার", "১ বর্গমিলিমিটার = ১ বর্গমিলিমিটার", _
    "১ বর্গমিলিমিটার = ১/৮৩৬১২৭.৩৬ বর্গগজ", "১ বর্গমিলিমিটার = ১/৯২৯০৩.০৪ বর্গফুট", "১ বর্গমিলিমিটার =
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
১/৬৪৫.১৬ বর্গইঞ্চি"}
    Dim CatAreaSqYardTo() As String = {"১ বর্গগজ = ১/৩০৯৭৬০০ বর্গমাইল", "১ বর্গগজ = ১/১১৯৫৯৯০.০৮৬৩ বর্গকিলোমিটার", "১ বর্গগজ = ১/১.১৯৫৯৯ বর্গমিটার", "১ বর্গগজ = ৮৩৬১২৭.৩৬ বর্গমিলিমিটার", _
    "১ বর্গগজ = ১ বর্গগজ", "১ বর্গগজ = ৯ বর্গফুট", "১ বর্গগজ = ১২৯৬ বর্গইঞ্চি"}
    Dim CatAreaSqFootTo() As String = {"১ বর্গফুট = ১/২৭৮৭৮৪০০ বর্গমাইল", "১ বর্গফুট = ১/১০৭৬৩৯১০.৪১৬৭ বর্গকিলোমিটার", "১ বর্গফুট = ১/১০.৭৬৩৯১০৪১৬৭ বর্গমিটার", "১ বর্গফুট = ৯২৯০৩.০৮ বর্গমিলিমিটার", _
    "১ বর্গফুট = ১/৯ বর্গগজ", "১ বর্গফুট = ১ বর্গফুট", "১ বর্গফুট = ১৪৪ বর্গইঞ্চি"}
    Dim CatAreaSqInchTo() As String = {"১ বর্গইঞ্চি = ১/৪০১৪৫০৫৬৫৮.০০৬৬ বর্গমাইল", "১ বর্গইঞ্চি = ১/১৫৫০০০৩১০০.০০৬২ বর্গকিলোমিটার", "১ বর্গইঞ্চি = ১/১৫৫০.০০৩১ বর্গমিটার", "১ বর্গইঞ্চি = ৬৪৫.১৬ বর্গমিলিমিটার", _
    "১ বর্গইঞ্চি = ১/১২৯৬ বর্গগজ", "১ বর্গইঞ্চি = ১/১৪৪ বর্গফুট", "১ বর্গইঞ্চি = ১ বর্গইঞ্চি"}
    '+++++++++++++ Block3 - Temperature to Temperature
    Dim CatTempCelciusTo() As String = {"১ ডিগ্রী.সেলসিয়াস = ১ ডিগ্রী.সেলসিয়াস", "১ ডিগ্রী.সেলসিয়াস = (°C × 180/100) + 32 ) ডিগ্রী.ফারেনহাইট", "১ ডিগ্রী.সেলসিয়াস = (°C + 273.15) কেলভিন"}
    Dim CatTempFahrenheightTo() As String = {"১ ডিগ্রী.ফারেনহাইট = (°F - 32) × 100/180 ডিগ্রী.সেলসিয়াস", "১ ডিগ্রী.ফারেনহাইট = ১ ডিগ্রী.ফারেনহাইট", "১ ডিগ্রী.ফারেনহাইট = (°F - 32) × 5 / 9 + 273.15 কেলভিন"}
    Dim CatTempKelvinTo() As String = {"১ কেলভিন = (K - 273.15) ডিগ্রী.সেলসিয়াস", "১ কেলভিন = (K × 9/5) − 459.67 ডিগ্রী.ফারেনহাইট", "১ কেলভিন = ১ কেলভিন"}
'++++++++++++++ 'Block4 -
    Dim CatWeightKilogramTo() As String = {"১ কিলোগ্রাম = ১ কিলোগ্রাম", "১ কিলোগ্রাম = ১/০.৯৩৩ সের", "১ কিলোগ্রাম = ১০০০ গ্রাম", "১ কিলোগ্রাম = ১০০০/১১.৬৬ ভরি"}
    Dim CatWeightSherTo() As String = {"১ সের = ০.৯৩৩১০ কিলোগ্রাম", "১ সের = ১ সের", "১ সের = ৯৩৩.১ গ্রাম", "১ সের = ৮০.০১৭১৫২৬৫৯ ভরি"}
    Dim CatWeightGramTo() As String = {"১ গ্রাম = ১/১০০০ কিলোগ্রাম", "১ গ্রাম = ১/৯৩৩.১০ সের", "১ গ্রাম = ১ গ্রাম", "১ গ্রাম = ১/১১.৬৬ ভরি"}
    Dim CatWeightVoriTo() As String = {"১ ভরি = ১/৮৫.৭৬৩২৯৩৩ কিলোগ্রাম", "১ ভরি = ১/৮০.০১৭১৫২৬৫৯ সের", "১ ভরি = ১১.৬৬ গ্রাম", "১ ভরি = ১ ভরি"}
    '+++++++++++ 'Block5
    Dim CatSurveytAcreTo() As String = {"১ একর = ১ একর", "১ একর = ১০০ শতাংশ", "১ একর = ৪৩৫৬০.১৭৮২ বর্গফুট"}
    Dim CatSurveyDecimalTo() As String = {"১ শতাংশ = ১/১০০ একর", "১ শতাংশ = ১ শতাংশ", "১ শতাংশ = ৪৩৫.৬০১৭৮২ বর্গফুট"}
    Dim CatSurveySqFootTo() As String = {"১ বর্গফুট = ১/৪৩৫৬০.১৭৮২ একর", "১ বর্গফুট = ১/৪৩৫.৬০১৭৮২ শতাংশ", "১ বর্গফুট = ১ বর্গফুট"}
```

উপরে ফাংশনের শুরুর পূর্বে এ্যারে ডিক্লারেশন কোড দেখানো হয়েছে। সবগুলো এ্যারের মধ্যে নির্দিষ্ট কয়েকটি ইলেমেন্ট আছে [০-৬ দু'টি (এতে আছে ৭টি করে টেক্সট ইলেমেন্ট), ০-২ দু'টি (এতে আছে ৩টি করে টেক্সট ইলেমেন্ট), ০-৩ একটি (এতে আছে ৪টি টেক্সট ইলেমেন্ট)]। এই ইলেমেন্টগুলো আমরা বাংলা ইউনিকোডে লিখেছি। এগুলো ব্যবহারকারী কম্বো বাক্স পরিবর্তন করলেই সম্পর্ক টেক্সট বাক্সে দেখতে পাবেন। সুতরাং কোন্ একক থেকে কোন্ এককে বদলানো হচ্ছে ও তাদের মধ্যে সম্পর্ক কী তা সাথে সাথে প্রদর্শিত হবে। পরের পৃষ্ঠাসমূহে প্রদর্শনের ফাংশন কোডগুলো দেখানো হয়েছে। ব্যাখ্যা পরবর্তীতে এসেছে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
'this function retuns  the correct Relation text for displaying in
TextBox3 -Formula or Relation- uses 3 integer params
Public Function UpdateRelationDisplay(SuplIndx1 As Integer,
SuplIndx2 As Integer, SuplIndx3 As Integer) As String
    Dim LoopCont As Integer 'to cotrol loop
    '============Length Start==========
    If SuplIndx1 = 0 And SuplIndx2 = 0 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenMileTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 1 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenKiloTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 2 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenMeterTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 3 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenMilliMeterTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 4 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenYardTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 5 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenFootTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 6 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatLenInchTo(LoopCont)
        End If
      Next
    End If
    '*****Length finished********
    '+++++++Area Start+++++++++
    If SuplIndx1 = 1 And SuplIndx2 = 0 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqMileTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 1 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqKiloTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 2 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqMeterTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 3 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqMilliMeterTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 4 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqYardTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 5 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqFootTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 6 Then
      For LoopCont = 0 To 6
        If SuplIndx3 = LoopCont Then
          Return CatAreaSqInchTo(LoopCont)
        End If
      Next
    End If
    '*****Area finished****
    '+++Temperature Start++
    If SuplIndx1 = 2 And SuplIndx2 = 0 Then
      For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
          Return CatTempCelciusTo(LoopCont)
        End If
      Next
    End If
    If SuplIndx1 = 2 And SuplIndx2 = 1 Then
      For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            Return CatTempFahrenheightTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 2 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatTempKelvinTo(LoopCont)
        End If
    Next
End If
'==========Temperature Finished========
'============Weight Start=======
If SuplIndx1 = 3 And SuplIndx2 = 0 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Rcturn CatWcightKilogramTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 1 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightSherTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightGramTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 3 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightVoriTo(LoopCont)
        End If
    Next
End If
'=========Weight Finished=======
'+++++++Survey Start+++++
If SuplIndx1 = 4 And SuplIndx2 = 0 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveytAcreTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 4 And SuplIndx2 = 1 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveyDecimalTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 4 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveySqFootTo(LoopCont)
        End If
    Next
End If
'=========Survey Finished=========
Return ""
End Function
'these two function gets the from unit word to display in textBox4 ans TextBox5
Public Function GetLeftWord(StrVar As String) As String
    Dim AllWords As String() = StrVar.Split(New Char() {" "})
    On Error GoTo LabelOne
    Return AllWords(1) 'this is the 2nd word which we need
LabelOne:
    Rcturn ""
End Function
'=========
Public Function GetRightWord(StrVar As String) As String
    Dim AllWords As String() = StrVar.Split(New Char() {" "}) 'Split is a function which seperates words
    Dim A As Integer = (AllWords.Length) 'get total number of words
    On Error GoTo LebelOne
    Return AllWords(A - 1) 'get the index of last word
LebelOne:
    Return ""
End Function
End Module
=============================================
```

এখানেই শেষ হলো মডিউলে ডিফাইন-করা ফাংশনের কোড। লক্ষ করুন, পুরো মডিউলে মোট তিনটি মাত্র ফাংশন আছে। প্রথম বড়ো ফাংশনের কাজ হলো প্রদর্শন ফিল্ডে দু'টি এককের সম্পর্ক রিটার্ন করা। দ্বিতীয় ও তৃতীয়টির কাজ হলো ফিল্ড ৪ ও ৫-এ সঠিক একক নাম প্রদর্শন করা।

প্রথম ফাংশনের মধ্যে মোট ২৪টি 'ইফ-এন্ড ইফ' [If - End If] টেস্ট আছে। এগুলো একই ধরনের। সুতরাং একটি সম্পর্কে পূর্ণ ব্যাখ্যাই যথেষ্ট হবে।

ফাংশনের 'ইফ-এন্ড ইফ' টেস্ট কোড রান করার আগে 'ফর-নেস্ট' [For - Next] লুপ নিয়ন্ত্রণের জন্য প্রথমেই আমরা একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি। এই একই ভেরিয়েবল ২৪টি টেস্টেই ব্যবহৃত হয়েছে। এবার প্রথম 'ইফ-এন্ড ইফ' ব্লকটি আবার তুলে ধরছি।

```
Dim LoopCont As Integer 'to control loop
    '============Length Start==========
    If SuplIndx1 = 0 And SuplIndx2 = 0 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenMileTo(LoopCont)
            End If
        Next
    End If
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

প্রথম লাইনেই LoopCont নামক ইন্টেজার ভেরিয়েবল ডিক্লারেশনটি দেখা যাচ্ছে। দ্বিতীয় লাইনে শুরু হয়েছে টেস্ট: If SuplIndx1 = 0 And SuplIndx2 = 0 Then -এর অর্থ হলো যদি ইনডেক্স১ [প্রথম কম্বো বাক্সের ইনডেক্স] ০ এবং ইনডেক্স২ [দ্বিতীয় কম্বো বাক্সের ইনডেক্স] ০ হয় তাহলে পরের লাইনে যাও- অন্যথায় নয়।

উভয় কন্ডিশন সত্য হলে পরের লাইনে এসে 'ফর-নেস্ট' লুপটি শুরু হবে। এ লুপের কাজ হলো তৃতীয় কম্বো বাক্সের ইনডেক্স কি তা সনাক্ত করা। সুতরাং ০ থেকে ৬ পর্যন্ত- এই মোট ৭ বার লুপ রান করবে। প্রত্যেকবার দেখে নেবে কম্বো বাক্স-৩ এর ইনডেক্সের সাথে লুপ সংখ্যার মিল আছে কি না। মিল হলেই মাঝের লাইনে যেয়ে রিটার্ন ভ্যালু প্রেরণ করে ফাংশনের কোড রান হওয়ার ইতি ঘটবে। তবে কী রিটার্ন করবে? রিটার্ন করবে একটি স্ট্রিং যা আসবে CatLenMileTo(LoopCont) থেকে। মনে আছে কি, আমরা প্রথমেই ২৪টি এ্যারে ভেরিয়েবল ডিক্লায়ার করেছিলাম? CatLenMileTo-হলো এই এ্যারেগুলোর মধ্যে প্রথমটির নাম। তৃতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা LoopCont-এর সমান। তাই এই ইনডেক্স মুতাবিক ঐ এ্যারের ইলেমেন্টই ফাংশন স্ট্রিং হিসেবে রিটার্ন করবে। কথাগুলো বুঝে না আসলে পুনরায় পাঠ করবেন।

প্রথম ব্লকের টেস্ট ফেইল করলে, দ্বিতীয় ব্লকে যেয়ে টেস্ট হবে- এভাবে যতক্ষণ কন্ডিশন মিলবে না ততক্ষণ টেস্ট করতে থাকবে। আর অবশ্যই, যে কোনো একটি টেস্টের ফলাফল সত্য হবে। কারণ, ব্যবহারকারী এই ২৪টি কম্বিনেশন ছাড়া বাইরে কিছু করতে পারবেন না। সুতরাং যেটাই টেস্ট থেকে সত্য ফলাফল দিক না কেনো, ফাংশন সঠিক টেক্সট রিটার্ন করবে। আর মনে রাখবেন, আমরা এই ফাংশনটি কল করেছি Form.vb এর ক্লাসের ভেতরস্থ ৩টি কম্বো বাক্স পরিবর্তন সিস্টেম ইভেন্ট সাব থেকে। অন্যকথায় ব্যবহাকারী এই তিনটি বাক্সের যে কোনটির ইনডেক্স পরিবর্তন করলেই ফাংশন কল করা হবে।

এবার শেষের দুটো ফাংশনের ওপর কিছুটা ব্যাখ্যার প্রয়োজন। এ দু'টো আবার দেখে নিই:

```
'Function uses StrVar as a string param- comes from calling sub
Public Function GetLeftWord(StrVar As String) As String
  'Split is a function which seperates words
  Dim AllWords As String() = StrVar.Split(New Char() {" "})
   On Error GoTo LabelOne
   Return AllWords(1) 'this is the 2nd word which we need
LabelOne:
    Return ""
 End Function
```

```
'Function uses StrVar as a string param- comes from calling sub
Public Function GetRightWord(StrVar As String) As String
   'Split is a function which seperates words
   Dim AllWords As String() = StrVar.Split(New Char() {" "})
   'get total number of words
  Dim A As Integer = (AllWords.Length)
  On Error GoTo LebelOne
  Return AllWords(A - 1) 'get the index of last word
LebelOne:
   Return ""
End Function
```

লক্ষ করুন, উভয় ফাংশনের একটি প্যারামিটার আছে। StrVar নামক এ প্যারামিটার আসবে যেখান থেকে এটি ডাক দেওয়া হয়েছে। অর্থাৎ আমরা সম্পর্ক বাক্সে প্রদর্শিত টেক্সট প্যারামিটার হিসাবে ব্যবহার করেছি। ফাংশন দু'টোর কাজ হলো বায়ের পূর্ণ শব্দ [যা হলো 'হতে' এককের নাম] এবং ডানের পূর্ণ শব্দ [যা হলো 'প্রতি' এককের নাম] বের করে রিটার্ন করা। প্রথম লাইনে 'AllWords' নামক একটি স্ট্রিং ভেরিয়েবল ডিক্লায়ার করেছি। সমান প্রতীকের ডানে প্যারামিটার স্ট্রিংকে 'Split(New Char() {" "})' নাম একটি সিস্টেম ফাংশন দ্বারা শব্দে শব্দে বিভক্ত করে নিয়েছি। এর পরের লাইনের কাজ হলো কোনো এরোর থাকলে অভিযোগ ছাড়াই ফাংশন থেকে বের হওয়ার ব্যবস্থা। অন্যথায় ফাংশন রিটার্ন করবে আমাদের কাঙ্ক্ষিত প্রথম শব্দটি [Return AllWords(1)]। দেখতেই পাচ্ছেন, উভয় ফাংশনের কাজ একই ধরনের। ডানেরটি রিটার্ন করবে শেষের শব্দটি [Return AllWords(A - 1)]। রিটার্ন হওয়া শব্দ দু'টো যথাক্রমে TextBox4 ও TextBox5-এ প্রদর্শিত হবে। সুতরাং আমাদের নিকট স্পষ্ট থাকবে কোন্ দু'টি একক নিয়ে কাজ হচ্ছে। শেষ হলো Public Module AllDisplayFunctionsModule -এর ব্যাখ্যা। এখন আমারেদকে বুঝতে হবে ক্যালকুলেশন মডিউলের কাজটি কিভাবে হয়।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

CalculationModule.vb-এ ডিফাইন করা ফাংশনগুলোর ওপর বিস্তারিত আলোচনা

ইতোমধ্যে যা কিছু কোড লিখিত হয়েছে তা সবই হলো ক্যালকুলেন সঠিকভাবে সম্পাদনের প্রস্তুতি মাত্র। অর্থাৎ কোন্ ফিল্ডে কি প্রদর্শন হবে, ব্যবহারকারী কোন্ ইউনিট থেকে কোন্ ইউনিটে পরিবর্তন করবেন, সংখ্যাটি কি, ইত্যাদি প্রশ্নের সঠিক জবাব আগের সকল কোড থেকে আমরা সুস্পষ্ট ও সঠিকভাবে সনাক্ত করেছি ও সংশ্লিষ্ট প্রদর্শন ফিল্ডে দেখানোরও ব্যবস্থা করেছি। এবার আসল কাজ তথা ক্যালকুলেশনের ক্রিয়াটি সম্পাদন করে রেজাল্ট দেখাতে হবে- আর এটাই তো পুরো প্রোগ্রামের আসল লক্ষ্য।

আমরা সিদ্ধান্ত করেছি, ক্যালকুলেন ও রেজাল্ট প্রদর্শন হবে একটি স্বয়ংক্রিয় কাজ। এর জন্য আলাদা কোনো বোতামে ক্লিক করার প্রয়োজন নেই। সুতরাং যে মুহূর্তে ব্যবহারকারী কোনো ক্রিয়া সম্পাদন করবেন- প্রোগ্রাম সাথে সাথে ফলাফল ও তথ্যাদি প্রদর্শন করবে- এটাই আমার চাই। আমরা জানি ব্যাবহারকারী মোট চারটি ক্রিয়ার যে কোনো একটি করতে পারেন। এগুলো হলো: ১. প্রথম কম্বো বাক্সের আইটেম পরিবর্তন [ক্যাটাগরি চয়েজ]। ২. দ্বিতীয় কম্বো বাক্সের আইটেম পরিবর্তন ['হতে' একক চয়েজ]। ৩. তৃতীয় কম্বো বাক্সের আইটেম পরিবর্তন ['প্রতি' একক চয়েজ]। ৪. সংখ্যা প্রবেশ ফিল্ড টেক্সট বাক্স-১ এ পরিবর্তন [সংখ্যা মান প্রবেশ]। আমাদের কোড এ চারটির মধ্যে কোন্‌টিতে রদবদল হয়েছে তা জেনে ক্যালকুলেশন করবে। এটা করতে যেয়ে আমরা পূর্বপ্রস্তুতি হিসেবে আগে ব্যাখ্যাত কোডগুলো কাজে লাগিয়েছি। এরপর যেখান থেকে যেভাবে প্রয়োজন CalculationModule.vb-এ ডিফাইন করা ফাংশনগুলোর সঠিকটি কল করে ফলাফল বের করে প্রদর্শন করেছি। কিভাবে? এক্ষুণি বুঝে নেবেন। তবে প্রথমে পুরো ক্যালকুলেন কোড ব্লকটি তুলে ধরা প্রয়োজন।

```
Public Module CalculationModule
    '========= Length Relation========
   Dim LengthRelation1() As Double = {1, 1.609344, 1609.344, 1609344, 1760, 5280, 63360}
    Dim LengthRelation2() As Double = {1 / 1.6093, 1, 1000, 1000000, 1093.6133, 3280.8399, 39370.0788}
    Dim LengthRelation3() As Double = {1 / 1609.3, 1 / 1000, 1, 1000, 1.0936133, 3.2808399, 39.3700788}
    Dim LengthRelation4() As Double = {1 / 1609344, 1 / 1000000, 1 / 1000, 1, 1 / 914.4, 1 / 304.8, 1 / 25.4}
    Dim LengthRelation5() As Double = {1 / 1760, 1 / 1093.6133, 1 / 1.0936, 914.4, 1, 3, 36}
    Dim LengthRelation6() As Double = {1 / 5280, 1 / 3280.8399, 1 / 3.28084, 304.8, 3, 1, 12}
    Dim LengthRelation7() As Double = {1 / 63360, 1 / 39370.0788, 1 / 39.37007884, 25.8, 1 / 36, 1 / 12, 1}
    '=========== Area Relation
    Dim AreaRelation8() As Double = {1, 2.59, 18683428.11, 2589988110336, 3097600, 27878400, 4014505658.0066}
    Dim AreaRelation9() As Double = {1 / 2.59, 1, 1000000, 1000000000000, 1195990.0463, 10763910.4167, 1550003100.0062}
    Dim AreaRelation10() As Double = {1 / 18683428.11, 1000000, 1, 1000000, 1.19599, 10.7639104167, 1550.0031}
    Dim AreaRelation11() As Double = {1 / 2589988110336, 1 / 1000000000000, 1 / 1000000, 1, 1 / 836127.36, 1 / 92903.04, 1 / 645.16}
    Dim AreaRelation12() As Double = {1 / 3097600, 1 / 1195990.0863, 1 / 1.19599, 836127.36, 1, 9, 1296}
    Dim AreaRelation13() As Double = {1 / 27878400, 1 / 10763910.4167, 1 / 10.7639104167, 92903.08, 1 / 9, 1, 144}
    Dim AreaRelation14() As Double = {1 / 4014505658.0066, 1 / 1550003100.0062, 1 / 1550.0031, 645.16, 1 / 1296, 1 / 144, 1}
    '=========== Weight Relation
    Dim WeightRelation15() As Double = {1, 1 / 0.933, 1000, (1000 / 11.66)}
    Dim WeightRelation16() As Double = {0.9331, 1, 933.1, 80.017152659}
    Dim WeightRelation17() As Double = {1 / 1000, 1 / 933.1, 1, 1 / 11.66}
    Dim WeightRelation18() As Double = {1 / (1000 / 11.6), 1 / 80.017152659, 11.66, 1}

    '=========== Survey Relation
    Dim SurveyRelation19() As Double = {1, 100, 43560.1782}
    Dim SurveyRelation20() As Double = {1 / 100, 1, 435.601782}
    Dim SurveyRelation21() As Double = {1 / 43560.1782, 1 / 435.601782, 1}
    'These functions returns Lengh Catagory to all combination calculated value. 2 parameters comes from calling sub
    Public Function CalculatedValueMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 6
            If ForCounter = ComBox3SelItemIndex Then
                Return LengthRelation1(ForCounter) * EnteredNumber
            End If
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueKiloMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation2(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation3(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueMilliMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation4(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueYardTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation5(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation6(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
Public Function CalculatedValueInchTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation7(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
'These functions returns Area Catagory to all combination calculated value. 3 parameters comes from calling sub
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Public Function CalculatedValueSqMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation8(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqKilometreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation9(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation10(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqMillimetre to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqMillimetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation11(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqYardTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation12(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqFt to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation13(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqInch to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqinchTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation14(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Temperature, celcius to all combination calculated value. 2 parameters comes from calling sub
Public Function CalculatedValueCelciusTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    If ComBox3SelItemIndex = 0 Then Return EnteredNumber * 1
    If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 1.8) + 32
    If ComBox3SelItemIndex = 2 Then Return EnteredNumber + 273.15
    Return 0
End Function
'This function returns Temperature, Fahrenheit to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueFahrenheitTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    If ComBox3SelItemIndex = 0 Then Return (EnteredNumber - 32) * (10 / 18)
    If ComBox3SelItemIndex = 1 Then Return EnteredNumber * 1
    If ComBox3SelItemIndex = 2 Then Return ((EnteredNumber - 32) * (5 / 9)) + 273.15
    Return 0
End Function
'This function returns Temperature, kelvin to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueKelvinTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    If ComBox3SelItemIndex = 0 Then Return (EnteredNumber - 273.15)
    If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 9 / 5) - 459.67
    If ComBox3SelItemIndex = 2 Then Return EnteredNumber * 1
    Return 0
End Function
'This function returns Weight, kg to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueKgTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 3
        If ForCounter = ComBox3SelItemIndex Then
            Return WeightRelation15(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Weight, sher to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSherTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 3
        If ForCounter = ComBox3SelItemIndex Then
            Return WeightRelation16(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function

'This function returns Weight, gram to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueGramTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 3
        If ForCounter = ComBox3SelItemIndex Then
            Return WeightRelation17(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'This function returns Weight, vori to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueVoriTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 3
            If ForCounter = ComBox3SelItemIndex Then
                Return WeightRelation18(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, acre to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueAcreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation19(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, decimal to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueDecimalTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation20(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, Sqft to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueSurveySqFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation21(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
End Module
```

এখানেই শেষ হলো `Public Module CalculationModule` -এর কোডগুলো। আমরা এবার কোডের ওপর বিস্তারিত ব্যাখ্যা তুলে ধরার প্রয়াস পাচ্ছি। এ ব্যাখ্যা শেষ হওয়ার সাথে একক পরিবর্তন প্রোগ্রাম সম্পর্কে তথ্য প্রদানের কাজটিরও ইতি ঘটবে। শিক্ষার্থীদের প্রতি পুনরায় আবদার জানাচ্ছি, পুরো প্রোগ্রাম সম্পর্কে যা কিছু আলোচনা হয়েছে এবং যেসব কোড তুলে ধরেছি তা ভালো করে বুঝার চেষ্টা করবেন। প্রোগ্রামের লজিক ও কোড সম্পর্কে যতো বেশি বুঝ আসবে, ততো বেশি লাভ হবে। এবার ব্যাখ্যার পালা।

প্রথমেই বলে রাখি, মডিউলের ভেতর মোট ২৪টি ছোট ফাংশন ডিফাইন করা হয়েছে। এর কারণ হলো, আমাদের প্রোগ্রাম মোট ২৪টি একক পরিবর্তনে সক্ষম। এগুলো হলো: ১. দূরত্ব ক্যাটাগরির ৭টি [ক. মাইল, খ. কিলোমিটার, গ. মিটার, ঘ. মিলিমিটার, ঙ. গজ, চ. ফুট ও ছ. ইঞ্চি]। ২. বর্গ ক্যাটাগরির ৭টি [ক. বর্গমাইল, খ. বর্গকিলোমিটার, গ. বর্গমিটার, ঘ. বর্গমিলিমিটার, ঙ. বর্গগজ, চ. বর্গফুট ও ছ. বর্গইঞ্চি]। ৩. তাপ ক্যাটাগরির ৩টি [ক. সেন্টিগ্রেড, খ. ফারেনহাইট ও গ. কেলভিন]। ৪. ওজন ক্যাটাগরির ৪টি [ক. কিলোগ্রাম, খ. সের, গ. গ্রাম ও ঘ. ভরি]। ৫. জমিরা মাপ ক্যাটাগরির ৩টি [ক. একর, খ. শতাংশ ও গ. বর্গফুট]।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

মডিউল ডিক্লারেশনের পর ও কোনো ফাংশন শুরুর আগেই আমরা বেশ কিছু গ্লোবাল ভেরিয়েবল ডিক্লায়ার করেছি। আমাদের ইচ্ছে এগুলো যে কোনো ফাংশনে ব্যবহার করা। লক্ষ করুন, সব ভেরিয়েবল হলো এ্যারে। এগুলোর টাইপও দশমিক সংখ্যা [Double]। চারটি ব্লকে এগুলো ডিক্লায়ার করেছি সহজে সনাক্তকরণের জন্য। যেমন: ব্লক ১. দূরত্ব সম্পর্কিত এ্যারে। ব্লক ২. বর্গ সম্পর্কিত এ্যারে। ব্লক ৩. ওজন সম্পর্কিত এ্যারে। ব্লক ৪. জমির মাপ সম্পর্কিত এ্যারে। লক্ষণীয়, তাপ সম্পর্কে কোনো এ্যারে ভেরিয়েবলের প্রয়োজন নেই। এর কারণ হলো, তাপমাত্রার পরিবর্তন হয় বিশেষ ফর্মুলা দ্বারা। কিন্তু বাকি সবই দু'টি সংখ্যার গুণফল।

গ্লোবাল এ্যারে ডিক্লারেশনের পর আমরা একে একে ২৪টি ফাংশন ডিফাইন করেছি। এ প্রসঙ্গে প্রথম ফাংশনটি কিভাবে কাজ করে তা বুঝিয়ে বলবো। এরপর তাপমাত্রা ক্যালকুলেশনের ফাংশন সম্পর্কে বলবো। কারণ এগুলো একটু ভিন্ন। আশাকরি শিক্ষার্থীদের জন্য এটুকু ব্যাখ্যাই যথেষ্ট হবে। প্রথমে একটি ফাংশন কোড আবার দেখে নিই:

```
'These functions returns Lengh Catagory to all combination calculated value. 2 parameters comes from calling sub
Public Function CalculatedValueMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation1(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
```

ব্যাখ্যা: প্রথমে জেনে রাখা দরকার যে, উক্ত ফাংশনটি কল করা হয়েছে ফর্মের ক্লাসের ভেতর ডিফাইন করা আরেকটি সাবরুটিন থেকে। সেখানে এই কোডটি দ্বারা ডাকা হয়েছে: `CalculatedValueMileTo(EnteredNumber, ComBox3SelItemIndex)`। লক্ষ করুন, কলিং কোডে ফাংশনের নাম ও প্যারামিটার একই। সুতরাং প্রোগ্রাম এই কোড ব্লক থেকে লাফ মেরে মডিউলের কোডে ডিফাইন করা একই নামের উক্ত ফাংশনে আসবে। অর্থাৎ উপরের ফাংশনটি এক্সেকিউট হবে। কিভাবে হবে? এক্ষুণি জেনে নিবেন।

প্রথম লাইনে আমরা একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি For-Next লুপটি চালানোর জন্য। এরপর সে লুপের ভেতর যেয়ে প্রয়োজনে মোট ৭বার পর্যন্ত [০-৬] ঘুরে দ্বিতীয় প্যারামিটর তথা `ComBox3SelItemIndex`-এ দেওয়া ইনডেক্স সংখ্যাটি খুঁজে বের করেছি। If ForCounter = ComBox3SelItemIndex কোডটি দ্বারা এ সন্ধান চালিয়েছি। কারণ প্রতিবার লুপ চলাকালে ForCounter -এর মান [০ থেকে ৬ পর্যন্ত] এক এক করে বাড়বে। আর দ্বিতীয় প্যারামিটামের মান অবশ্যই ০ থেকে ৬ পর্যন্ত যে কোনো একটি সংখ্যা হবেই। উভয় সংখ্যা সমান হলেই আমরা বুঝতে পারলাম কোন্ হিসাবটি দরকার। সুতরাং Return LengthRelation1(ForCounter) * EnteredNumber-দ্বারা সঠিক ফলাফল পাঠিয়ে দিচ্ছি কলিং সাবরুটিনে। এখানে লক্ষণীয়, হিসাব করতে দু'টি সংখ্যার প্রয়োজন। এর প্রথমটি আসবে গ্লোবেল ভেরিয়েবল দূরত্ব সম্পর্ক LengthRelation1 -থেকে। ভেরিয়েবলের এ্যারেতে মোট ৭টি সংখ্যা আছে। সঠিকটি সনাক্ত করে আনার জন্য ইনডেক্স প্রয়োজন। এই ইনডেক্সই হলো দ্বিতীয় প্যারামিটারের সমান [লুপ থেকে প্রাপ্ত] ForCounter এর মানটি। দ্বিতীয় সংখ্যাটি আসবে প্রথম প্যারামিটার থেকে- যা কলিং সাবই নির্ধারণ করে দিয়েছে। অর্থাৎ যে সংখ্যাটি ব্যবহারকারী টেক্সট ফিল্ডে ঢুকিয়েছেন সেটাই হলো EnteredNumber-এর মান। যদি কোনো কারণে সব ফেল করে তাহলে ফাংশন ০ রিটার্ন করবে। বাস্তবে সঠিকভাবে সংখ্যা ঢুকালে ফাংশন থেকে সর্বদাই সঠিক হিসাব রিটার্ন হবে। আগেই বলেছি বাকি প্রায় সবগুলো ফাংশনই একইভাবে কাজ করে। সুতরাং ওগুলো সম্পর্কে আর ব্যাখ্যার প্রয়োজন নেই। শিক্ষার্থীরা কিন্তু সব কোড দেখে নিবেন ও কিভাবে কাজ করে বুঝার চেষ্টা করবেন। শেষের দিকে ডিফাইন করা তাপমাত্রার ফাংশন সম্পর্কে কিছু ব্যাখ্যার প্রয়োজন আছে। সুতরাং পরের পৃষ্ঠায় আমরা এ সম্পর্কে তথ্যাদি তুলে ধরার প্রয়াস পাচ্ছি।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

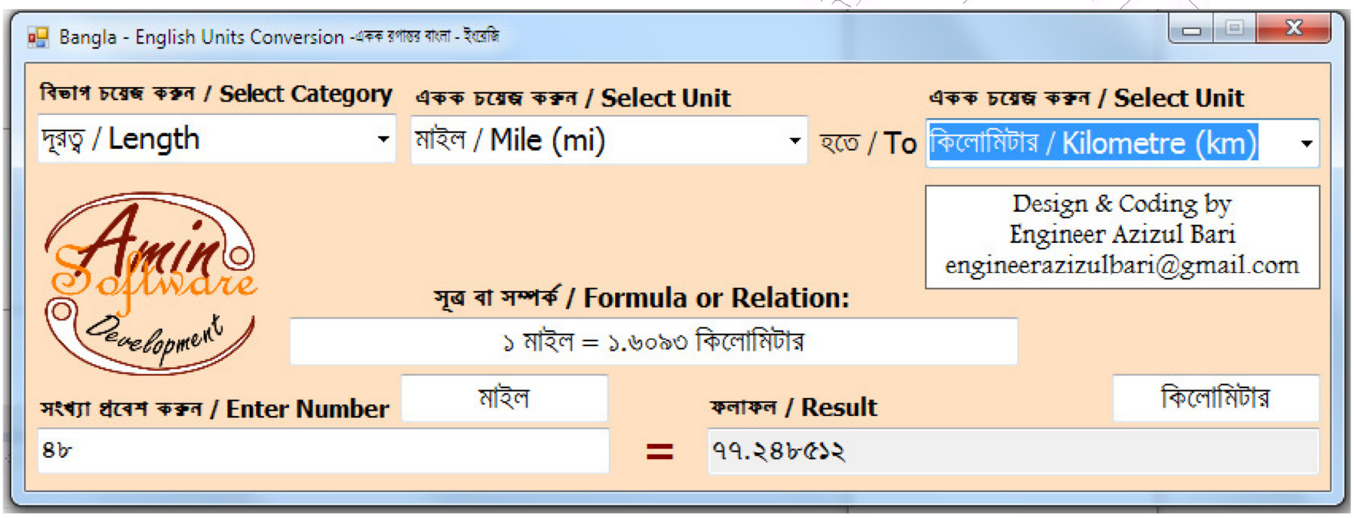
```
'This function returns Temperature, celcius to all combination calculated value. 2 parameters comes from calling sub
Public Function CalculatedValueCelciusTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    If ComBox3SelItemIndex = 0 Then Return EnteredNumber * 1
    If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 1.8) + 32
    If ComBox3SelItemIndex = 2 Then Return EnteredNumber + 273.15
    Return 0
End Function
```

উপরে তাপ ক্যাটাগরির প্রথম ফাংশন কোড আবার তুলে ধরেছি। ফাংশনের ভেতর ৪টি মাত্র লাইন আছে। প্রথম তিনটি দ্বারাই আমরা সঠিক হিসাব করে কলিং সাবে ফলাফল ফেরৎ পাঠিয়েছি। যেমন: লাইন ১: এতে টেস্ট করেছি দ্বিতীয় প্যারামিটার এর মান ০ কি না? যদি হয়ে থাকে তাহলে ১ এর সঙ্গে প্রথম প্যারামিটার পূরণ হবে। অনুরূপ লাইন ২: দ্বিতীয় প্যারামিটারের মান ১ থাকলে সংখ্যা নির্ণিত হবে দেখানো ফর্মুলা মুতাবিক। লাইন ৩: দ্বিতীয় প্যারামিটারের মান ২ হলে প্রথম প্যারামিটারের সাথে ২৭৩.১৫ যোগ হবে। লক্ষ্য করুন, তিনটির মধ্যে শুধুমাত্র একটি মান সত্য। সুতরাং ফাংশন থেকে সত্য মানের সংখ্যা রিটার্ন হবে প্রতিবারই। আর কোনো কারণে সমস্যা হলো শেষের লাইনটি এক্সেকিউট হবে।

এ হচ্ছে ‘Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার’ - নামক আমাদের সৃষ্ট নতুন পিসি এ্যাপলিকেশনের কোডসহ পূর্ণাঙ্গ ব্যাখ্যা। কোডগুলো পেতে আমার নিকট ই-মেইল করুন এই ঠিকানায়: **engineerazizulbari@gmail.com**। আপনার ই-মেইলে কোড পাঠিয়ে দেবো। ফলে সময় ও কষ্টের বিনিময়ে এতো কোড নিজে টাইপ করা থেকে বিরত থাকতে পারেন। এছাড়া থাকতে পারেন ভুল-ত্রুটি থেকে নিরাপদ।

টেস্ট ও বিস্তারণ

বহু কষ্ট, সময় ক্ষেপণ ও ধৈর্যের বিনিময়ে আমরা তৈরি করেছি ‘Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার’ নামক এ প্রোগ্রামটি। কোডিংয়ের সময় আপনাকে বার বার রান করে দেখতে হবে সবকিছু সঠিকভাবে কাজ করছে কি না। তবে যেহেতু এখন পুরো প্রোগ্রামের কাজ শেষ হয়ে গেছে তাই ফাইন্যাল টেস্ট করা জরুরী। আমরা চাই সকলেই এই প্রোগ্রাম থেকে উপকৃত হবে। সুতরাং ইন্টারনেট বা অন্য কোনো উপায়ে প্রোগ্রামটি পাবলিশ বা প্রকাশ করতে হবে। এ কাজের নামই বিস্তারণ। প্রথমে নিচের ছবিটি দেখুন।



উপরের চিত্রে আমরা মাইল থেকে কিলোমিটার একক পরিবর্তন করেছি। ৪৮ মাইলে হয় ৭৭.৬০৯৩ কি.মি। আমরা সব ফাংশন পরীক্ষা করে দেখেছি। প্রোগ্রাম সঠিক ফলাফল দিয়েছে। কোনো সমস্যা পরিলক্ষিত হয় নি। সুতরাং আমরা সফল! এবার এটি কিভাবে বিস্তারণ করবো তা জেনে নিই। কেমন?